



①⑨ BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENTAMT

⑫ **Offenlegungsschrift**
⑩ **DE 196 31 005 A 1**

⑤① Int. Cl.⁸:
G 01 R 31/3167

DE 196 31 005 A 1

②① Aktenzeichen: 196 31 005.9
②② Anmeldetag: 1. 8. 96
④③ Offenlegungstag: 6. 2. 97

③① Unionspriorität: ③② ③③ ③①
01.08.95 US 510397

⑦① Anmelder:
Schlumberger Technologies Inc., San Jose, Calif.,
US

⑦④ Vertreter:
Sparing . Röhl . Henseler, 40237 Düsseldorf

⑦② Erfinder:

Rosenthal, Daniel, Saratoga, Calif., US; Konath,
Kannan, Fremont, Calif., US; Whyte, Robert,
Ringwood, Hants, GB; Norton, Eric, Cupertino, Calif.,
US; Pearce, Stuart Robert, San Jose, Calif., US

⑤④ Analogkanal für Mischsignal-VLSI-Tester

⑤⑦ Mischsignal-Tester-Architektur wird vorgeschlagen, welche den Transfer von Daten minimiert, parallele Datennachbearbeitung innerhalb der Analogkanäle anbietet und flexible Synchronisation ermöglicht. Mehrere Analogkanäle, von denen jeder einen Quellendigitalsignalprozessor (DSP), einen digitalen Quellensequenzler, eine digitale Quelleninstrumentierung, analoge Quelleninstrumentierung, analoge Meßinstrumentierung, digitale Meßinstrumentierung, einen Digitalstiftmultiplexer, einen Digitalmeßsequenzler, DSP-adressierbare Mehrbanken-Einfangspeicher, einen Einfangdigitalsignalprozessor und einen Inter-DSP-Rückkopplungspfad für Kommunikation zwischen dem Quellen-DSP und dem Einfang-DSP umfaßt. Jeder Analogkanal kann in einer Rückkopplungsschleife durch entweder seine analoge und/oder digitale Instrumentierung unter Verwendung des Inter-DSP-Rückkopplungspfad angeordnet sein. Die Reaktion des Prüflings wird in dem Kanal verarbeitet, das Resultat wird verwendet zum Definieren von Parametern für den nachfolgenden Testzyklus und ein Signal entsprechend diesen Parametern wird erzeugt und an den Prüfling angelegt. Dadurch wird die Testgeschwindigkeit erhöht. Der Quellen-DSP kann Signale in Realzeit synthetisieren und diese an den Prüfling durch analoge oder digitale Quelleninstrumentierung anlegen, und kann Quellensequenzerspeicheradressen synthetisieren (also Zeiger zu Wellenformdaten, die im Speicher abgelegt sind, welche Wellenformen oder Wellenformsegmente repräsentieren), und ...

DE 196 31 005 A 1

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

BUNDESDRUCKEREI 12. 96 602 066/653

32/24

Die vorliegende Erfindung bezieht sich auf Verfahren und Vorrichtungen für das Testen von Mischsignal-VLSI-Komponenten.

Digitale integrierte Schaltkreiskomponenten (IC) werden typischerweise getestet, indem man an Stifte der Komponente ein Muster binärer Stimuliersignale in vorbestimmten Muster- und Zeitlagebeziehungen anlegt. Das digitale Testsystem beobachtet die resultierenden digitalen Ausgangssignale der Komponente und vergleicht sie mit einer vorher definierten Wahrheitstabelle. Eine Gut- oder Versagensentscheidung resultiert abhängig davon, ob die Bits (1er und 0en) an den Ausgangsstiften der Komponente den Bits der Wahrheitstabelle in jedem Zeitintervall entsprechen. Testsysteme für solche Komponenten sind flexibel und programmierbar, um die Erfordernisse der Type von zu testender Komponente zu erfüllen. Ein Beispiel eines hochgeschwinden programmierbaren digitalen Testsystems ist das ITS 9000FX-System, das im Handel von Schlumberger Technologies, San Jose, Kalifornien, erhältlich ist.

Andere zu testende Komponenten sind nicht rein digital. Diese Komponenten, die als Mischsignalkomponenten bekannt sind, können sowohl Digital- als auch Analogsignalcharakteristiken haben. Mischsignalkomponenten sind häufig überwiegend digital, können jedoch nicht so getestet werden, wie man rein digitale Komponenten testet. Mischsignalkomponenten können Stifte haben, die einen oder mehrere Analogsignaleingänge benötigen (beispielsweise Analog-Digital-Umsetzer (ADC's) oder einen oder mehrere Analogsignalausgänge haben, beispielsweise Digital-Analog-Umsetzer (DAC's) zusätzlich zu Digitalsignaleingängen oder -ausgängen. Mischsignalkomponenten können Stifte aufweisen, die digitale Repräsentationen von Analogsignalen empfangen oder liefern (beispielsweise Codier-Decodier-Komponenten (Codecs)). Digitale Repräsentationen von Analogsignalen unterscheiden sich von Digitalsignalen, indem Information, die in digitaler Form codiert ist, analoge Werte repräsentiert. Es genügt nicht, die Ausgangsbits der codierten Signale mit einer vorbestimmten Wahrheitstabelle innerhalb eines Zeitfensters zu vergleichen, weil es die in den 1ern und 0en codierte Information ist, die abgeschätzt werden muß um festzustellen, ob die Komponente so arbeitet, wie sie sollte. Das codierte Signal kann in der Form von seriellen Daten auf einem Stift oder parallelen Daten auf mehreren Stiften vorliegen und kann codiert sein unter Verwendung irgendeiner Art von Schemata. Zusätzlich zum Testen der Gleichstromcharakteristiken (DC) der Komponente muß der Tester als akzeptabel jede Zahl von unterschiedlichen Bitkombinationen erkennen, welche im wesentlichen denselben Analogwert innerhalb eines bestimmten Fehlerbandes repräsentiert. Um den Ausgang des Prüflings zu analysieren, wird Digitalsignalverarbeitung verwendet, um die quantitativen Verhaltensparameter von den analogen und digitalen Ausgangssignalen zu extrahieren.

Das Testen von Mischsignalkomponenten ist zeitaufwendig. Ein einzelner Testzyklus kann das Anlegen eines Satzes von Eingangsstimuli an die Komponente und Messen der Reaktion der Komponente umfassen. Beispielsweise wird eine Analogspannung an einen ADC angelegt und der resultierende digitale Ausgang wird erfaßt. Der Testzyklus wird für viele unterschiedliche Sätze von Stimuli wiederholt zum Abschätzen des Verhaltens der Komponente unter verschiedenen Bedin-

gungen. Beispielsweise werden an einen ADC Analogspannungen über dem erwarteten Betriebsbereich angelegt. Wenn das Signal-Rausch-Verhältnis der Messung niedrig ist, können vielfache Tests auszuführen sein für jeden Satz von Eingangsstimuli und die Resultate gemittelt werden. Die Wiederholbarkeit des Komponentenverhaltens kann gegebenenfalls auch zu testen sein, was noch mehr Testzyklen benötigt.

Die an den Prüfling anzulegenden Stimuli hängen oft von seiner Reaktion auf Stimuli ab, die in einem vorhergehenden Testzyklus angelegt worden waren. Die Nachbearbeitung muß deshalb schnell erfolgen, wenn die Gesamttestzeit innerhalb vernünftiger Grenzen gehalten werden soll.

Es sind konventionelle Mischsignaltestsysteme bekannt, bei denen ein einziger Wirtscomputer den gesamten Testprozeß steuert und auch die digitale Signalverarbeitung für mehrere Analogkanäle ausführt. Daten von mehreren Quellen werden synchron mit einem Haupttestprogramm verarbeitet. In einigen Testsystemen wird der Wirtscomputer durch einen Matrixprozessor oder digitalen Signalprozessor ergänzt. Selbst wenn schnelle Fourier-Transformationsverarbeitung (FFT) in jedem Analogkanal ausgeführt wird, haben Testsysteme, bei denen sich die Kanäle Rechenressourcen teilen, inhärente Nachteile. Zunächst muß eine große Menge an Information durch jeden Analogkanal laufen, der Signale dem Prüfling liefert oder von ihm empfängt. Signale von dem Prüfling werden digitalisiert und im Speicher in dem Kanal eingefangen. Diese große Datenmenge muß über einen Bus zu dem von den Kanälen geteilten Prozessor transferiert werden, bevor die Nachbearbeitung beginnen kann. Diese Datentransferverzögerung wird signifikant, wenn sie multipliziert wird mit der Anzahl von Kanälen und mit der Anzahl von Testzyklen, die am Prüfling auszuführen sind. Zweitens muß der Transfer der Daten über einem gemeinsamen Bus zu dem von den Kanälen geteilten Prozessor sequentiell Kanal um Kanal erfolgen. Die Daten werden dann nachbearbeitet in dem von den Kanälen geteilten Prozessor und/oder Matrixprozessor sequentiell Kanal um Kanal. Sequentieller Transfer und Verarbeitung bewirken in gleicher Weise eine Durchsatzverzögerung. Die Testgeschwindigkeit wird durch die Architektur des Testsystems beschränkt.

Zusätzlich haben Systeme nach dem Stand der Technik einen einzigen "Programmfaden" — das Hauptprogramm. Das heißt, das Hauptprogramm führt das Einfangen und dann die Nachbearbeitung von Daten mittels des auf die Kanäle verteilten Prozessors aus. Verarbeitungen, verzweigt zu einem von mehreren Kanälen benutzten Matrixprozessor oder Datensignalprozessor (DSP) laufen nicht parallel mit dem Hauptprogramm. Demgemäß wird die DSP-Hardware nicht in optimaler Weise in den Systemen nach dem Stand der Technik benutzt, weil das System keine tatsächlich asynchrone Steuerung der DSP-Hardware erlaubt.

Mischsignalkomponenten, die sowohl analoge als auch digitale Signale verarbeiten, haben größere Funktionalität, Verhaltensgüte und Geschwindigkeit als jemals. Diese Komponenten müssen als ein System in ihrer Arbeitsreihenfolge getestet werden mit kombiniertem Testen von digitalen und analogen Schaltkreisen einschließlich der Gleichstromcharakteristiken. Um eine Mischsignalkomponente als ein System zu testen, erfordert die Erzeugung und die Messung von analogen und digitalen Signalen Eingang zu bzw. Ausgang von der Komponenten in flexibler Synchronisation. Die ver-

fügbaren Tester sind nicht adäquat, um synchrone und asynchrone Steuerung von Mischsignalkomponenten zu ermöglichen. Schnellere und flexiblere Testsysteme für Mischsignalkomponenten sind wünschenswert.

Der Patentanspruch 1 definiert die Vorrichtung gemäß der Erfindung.

Gemäß bevorzugten Ausführungsformen der Erfindung werden eine Systemarchitektur und ein entsprechendes Verfahren vorgeschlagen, durch welche der Transfer von Daten minimiert wird, parallele Datennachbearbeitung angeboten wird innerhalb des Analogkanals und flexible Synchronisation ermöglicht wird.

Eine Mehrzahl von Analogkanälen ist vorgesehen, jeder mit einem Quellendigitalsignalprozessor, einem Datenquellensequenzler, Digitalquelleninstrumentierung, Analogquelleninstrumentierung, Analogmeßinstrumentierung, Digitalmeßinstrumentierung, einem digitalen Stütmultiplexer, einem digitalen Meßsequenzer, DSD-adressierbarem Mehrbankeinfangspeicher, einem Einfangdigitalsignalprozessor und einem Inter-DSP-Rückkopplungspfad für die Kommunikation zwischen dem Quellen-DSP und dem Einfang-DSP. Jeder Analogkanal kann in einer vollständigen Rückkopplungsschleife entweder über seine analoge oder digitale Instrumentierung angeordnet sein oder eine Kombination derselben unter Verwendung des Inter-DSP-Rückkopplungspfades.

Die Reaktion des Prüflings wird in dem Kanal verarbeitet, das Ergebnis der Verarbeitung wird verwendet zur Definition von Parametern für einen nachfolgenden Testzyklus, und ein Signal entsprechend diesen Parametern wird erzeugt und an den Prüfling angelegt. Die Möglichkeit, das Ergebnis eines Testzyklus auf diese Weise innerhalb des Analogkanals rückzuführen, um so den nächsten Testzyklus zu definieren, beschleunigt den Testablauf. Der Quellen-DSP kann Signale in Realzeit synthetisieren und diese an den Prüfling durch die analoge oder digitale Quelleninstrumentierung anlegen. Der Quellen-DSP kann Quellensequenzerspeicheradressen synthetisieren (Zeiger zu Wellenformdaten, abgespeichert im Speicher, welche Wellenformen oder Wellenformsegmente repräsentieren), und zwar in Realzeit und diese Signale an den Prüfling durch die analoge oder digitale Quelleninstrumentierung anlegen.

Die Reaktion des Prüflings wird in den Einfangspeicher in dem Kanal eingeschrieben, der direkt durch den Einfang-DSP adressierbar ist, anstatt daß ein zeitweiliger Pufferspeicher durchlaufen wird. Das Vermeiden des Transfers von Daten vor der Verarbeitung beschleunigt den Testablauf weiter. Der Mehrbankeinfangspeicher unter Steuerung des Einfang-DSP ermöglicht, daß Daten, welche die Reaktion des Prüflings repräsentieren, in eine Bank eingeschrieben werden, während vorher geschriebene Daten in einer anderen Bank verarbeitet werden. Die Verschachtelung von Dateneinfang und Datenverarbeitung erlaubt auf diese Weise, daß der Dateneinfang und die Datenverarbeitung gleichzeitig ablaufen, was den Testablauf weiter beschleunigt.

Mit Quellen- und Einfang-DSP's für jeden Analogkanal wird eine "Verkettungstechnik" verwendet, bei der jeder eine Anzahl von Verarbeitungsketten einen Prozeß unabhängig von den anderen Ketten ausführt zwecks optimaler Ausnutzung der Ressourcen. Die Hauptkette des Testablaufs ist nicht beschränkt durch die Notwendigkeit der häufigen Kommunikation mit den DSP's und kann verschiedene andere Aufgaben erfüllen (Berechnung, Pufferverwaltung, digitale Tests, die

keine Analogkanal-Hardware involvieren, etc.), nachdem verschiedene auf den DSP's der Analogkanäle auszuführende Prozesse verteilt worden sind. Da jeder Analogkanal des Testsystems selbstlaufend ist (d. h. die Analogkanäle teilen sich nicht die Verarbeitungsressourcen), können mehrere Mischsignalmessungen parallel ausgeführt werden, ohne Verschlechterung der Testzeit. Kanäle können einzeln oder in Sätzen programmiert werden zum Erleichtern schneller Testzeiten für das parallele Testen. Analoge Taktgeber erzeugen hochgenaue hochauflösende Taktsignale mit niedrigem Phasenrauschen, die mit einem Hauptsystemtaktgeber (und demgemäß mit dem digitalen Untersystem) phasenverriegelt sind, um die DSP-Techniken zu vereinfachen.

Die Verarbeitungen sind unabhängig, können sich jedoch bei Erfordernis Daten teilen. Ein Verarbeitungsmanager verfolgt die verschiedenen auszuführenden Tests. Die Synchronisierung von Prozessen wird wirksam erreicht bei einem vorbestimmten Zeitpunkt, wenn Entscheidungen zu treffen sind oder Abhängigkeiten auftreten. Dieser Ansatz stellt den optimalen Einsatz des DSP für den Dateneinfang und die Nachbearbeitung sicher. Es gibt keine Latenzzeit, die erforderlich wäre für die Bewegung der eingefangenen Daten von dem Einfangspeicher zu dem Einfang-DSP. Da der Einfang-DSP direkten Zugriff auf den Einfangspeicher hat, brauchen die eingefangenen Daten nicht von dem Einfangspeicher zu einem gesonderten DSP-(oder Matrixprozessor-) Speicher vor Beginn der Verarbeitung überführt zu werden.

Diese und andere Merkmale der Erfindung sind in größeren Einzelheiten nachstehend unter Bezugnahme auf die beigefügten Zeichnungsfiguren erläutert.

Fig. 1 illustriert einige Funktionsmöglichkeiten eines Mischsignal-Testers gemäß der Erfindung;

Fig. 2 zeigt eine Architekturübersicht eines Mischsignal-Testsystems gemäß der Erfindung;

Fig. 3 zeigt ein Blockdiagramm der Signalkanäle eines Mischsignal-Testsystems gemäß einer bevorzugten Ausführungsform der Erfindung auf hohem Niveau;

Fig. 4 ist ein mehr ins einzelne gehendes Blockdiagramm von Signalfpfaden des Testsystems der Fig. 3;

Fig. 5 zeigt die Architektur der DSP-Hardware von Fig. 4 auf hohem Niveau;

Fig. 6 und 7 zeigen den Analogkanal der Fig. 4 im Blockmodusbetrieb;

Fig. 8 ist ein Blockdiagramm zur Darstellung der Gesamtarchitektur eines Mischsignal-Testsystems gemäß der Erfindung;

Fig. 9 ist ein Flußdiagramm eines Zeitzuteilprogramms, das in dem Testerprozessor läuft für die Steuerung des Hochpegelbetriebs eines Testsystems gemäß der Erfindung;

Fig. 10 ist eine funktionelle Wiedergabe einer Benutzerschnittstelle für die Konfiguration eines Testsystems gemäß der Erfindung;

Fig. 11 ist ein Display einer weiteren Benutzerschnittstelle für die Konfiguration eines Testsystems gemäß der Erfindung;

Fig. 12 zeigt eine Menü-Hierarchie von operationellen DSP-Funktionen eines Testsystems gemäß der Erfindung;

Fig. 13 ist ein Display einer Benutzerschnittstelle, die Blöcke aufweist zum Veranlassen der Eingabe von Parametern zwecks Aufstellung eines Analogkanals für hochgenaue Wellenformmessung gemäß der Erfindung;

Fig. 14 ist ein Display einer Benutzerschnittstelle mit Blöcken zum Einleiten der Eingabe von Parametern

zwecks Aufstellung eines Analogkanals für hochgenaue Wellenformzulieferung gemäß der Erfindung;

Fig. 15 bis 20 zeigen graphische Displays zur Illustration von Merkmalen von Software-Werkzeugen, brauchbar bei der Steuerung eines Testers gemäß der Erfindung;

Fig. 21 zeigt ein Zustandsdiagramm für den Laufzeitbetrieb eines Mischsignal-Testers gemäß der Erfindung;

Fig. 22 zeigt Betriebssequenzen eines Mischsignal-Tests gemäß der Erfindung;

Fig. 23 zeigt ein "Lade"-Unterzustandsdiagramm für die Konfiguration von Analogelementen eines Mischsignal-Testers gemäß der Erfindung;

Fig. 24 zeigt ein "Installations"-Unterzustandsdiagramm für die Initialisierung von Hardware-Elementen eines Mischsignal-Testers gemäß der Erfindung;

Fig. 25 zeigt ein "Initialisations"-Unterzustandsdiagramm für die Initialisierung von Analogelementen eines Mischsignal-Testers gemäß der Erfindung;

Fig. 26 zeigt ein erstes "Exekutions"-Unterzustandsdiagramm von Analogelementen bei der Ausführung eines Tests in einem Mischsignal-Tester gemäß der Erfindung.

Fig. 27 zeigt ein zweites "Exekutions"-Unterzustandsdiagramm von Analogelementen bei der Ausführung eines Tests in einem Mischsignal-Tester gemäß der Erfindung;

Fig. 28 zeigt ein drittes "Exekutions"-Unterzustandsdiagramm von Analogelementen bei der Ausführung eines Tests in einem Mischsignal-Tester gemäß der Erfindung;

Fig. 29 zeigt ein viertes "Exekutions"-Unterzustandsdiagramm von Analogelementen bei der Ausführung eines Tests in einem Mischsignal-Tester gemäß der Erfindung;

Fig. 30 zeigt Hauptsignaloperationen eines Codec-Sende-, Empfangssignal-Rauschverhältnistests gemäß der Erfindung;

Fig. 31 zeigt Hauptsignaloperationen eines Modem-Bit-Fehlerratetests gemäß der Erfindung; und

Fig. 32 zeigt Hauptsignaloperationen eines A/D-Regelschleifen-Codeflankenübergangs-Genauigkeitstests gemäß der Erfindung.

Fig. 1 illustriert einige Funktionsmöglichkeiten eines Mischsignal-Testers gemäß der Erfindung. Ein Mischsignal-Prüfling 100 kann einen Analogeingang 105 haben, an den ein analoges Stimuliersignal spezifizierten Formats anzulegen ist, beispielsweise eine Sinuswelle 110. Das analoge Stimuliersignal wird als Digitaldaten repräsentiert, die von einem Datensequenzner 115 geliefert werden und in Analogform durch einen Digital-Analog-Umsetzer 120 gebracht werden. Das Analogsignal wird dann dem Eingang 105 über ein Filter 125 zugeführt. Ein Mischsignal-Prüfling 100 kann einen digitalen Eingang 130 haben, an den analoge Information codiert als digitale Daten spezifizierten Formats zu liefern sind. Die Digitaldaten werden von einem Datensequenzner 135 an einen Formatierer 140 geliefert, der das formatierte Signal dem Digitaleingang 130 zuführt.

Ein Mischsignal-Prüfling kann einen analogen Ausgang 145 haben, der ein Analogsignal 150 ausgibt, das von dem Testsystem zu analysieren ist. Das Signal läuft durch ein Filter 155 zu einem Analog-Digital-Umsetzer 160. Das resultierende digitalisierte Signal wird im Eingangsspeicher 165 gespeichert, auf den ein digitaler Signalprozessor DSP 170 Zugriff hat. Ein Mischsignal-Prüfling kann einen digitalen Ausgang 175 haben, der eine Analoginformation repräsentierendes Digitalsi-

gnal abgibt. Das Digitalsignal gelangt zu einem Formatierer 180, der das Digitalsignal decodiert und die resultierende decodierte Information im Eingangsspeicher 185 abspeichert. Daten im Eingangsspeicher 185 sind für einen DSP 190 zugänglich. DSP 170 und DSP 190 sind programmierbar für die Ausführung von zeitbasierten und frequenzbasierten Analysen der von dem Prüfling empfangenen Signale.

Fig. 2 zeigt eine Architekturübersicht eines Mischsignal-Testsystems gemäß der Erfindung. Eine Arbeitsstation 200 bildet eine Benutzerschnittstelle für Programmieren und Steuerung des Testers auf oberstem Niveau und kommuniziert mit einem Testercomputer 210, der seinerseits mit DSP-Prozessoren 220 kommuniziert. Ein Paar von DSP-Prozessoren ist für jeden Analogkanal vorgesehen, wie in größeren Einzelheiten unten erläutert. DSP-Prozessoren 220 treiben Analogsequenzer 230, welche Taktsignale nach Bedarf von Analogtaktquellen 240 erhalten. Analogtaktquellen 240 sind auf einen digitalen Haupttaktgeber 250 bezogen. Digitale Stiftscheiben 260 kommunizieren mit digitalen Stiftelektroniken 270. Analoge Sequenzer 230 kommunizieren mit analogen Stiftelektroniken 280. Digitale Stiftelektroniken 270 und analoge Stiftelektroniken 280 liefern Stimuliersignale an die Stifte des Prüflings 290 und empfangen Reaktionssignale von diesen. Die analogen und digitalen Signale werden durch Synchronisation von analogen Sequenzern 230, Analogkanal-Taktsignalgeneratoren 240, digitalem Hauptsignaltaktgenerator 250 und digitalen Stiftscheiben 260 koordiniert.

Fig. 3 zeigt ein Blockdiagramm der Digitalkanäle 300 und Analogkanäle 305 eines Mischsignal-Testsystems auf hohem Niveau gemäß einer bevorzugten Ausführungsform der Erfindung. Die Anzahl von Digitalkanälen 300 und Analogkanälen 305 ist eine Sache der Auslegung; eine bevorzugte Ausführungsform hat 448 Digitalkanäle und vier Analogkanäle. Der steuerbare Haupttaktgeber 250 liefert ein Taktsignal bei beispielsweise 306,5 MHz bis 312,5 MHz an die digitalen Stiftscheiben (digitalen Stiftsteuerer) jedes Digitalkanals und an den analogen Taktgenerator jedes Analogkanals, beispielsweise an den digitalen Stiftsteuerer 310 des Digitalkanals 1, den digitalen Stiftsteuerer 315 des Digitalkanals 448 und an die Analogkanaltaktsignalgeneratoren 240.

Die Digitalstiftsteuerer kommunizieren mit dem Prüfling über Stiftelektronikkarten (PE) und eine Lastplatine 320, auf der der Prüfling montiert ist; beispielsweise kommunizieren die digitalen Stiftsteuerer 310 und 315 mit dem Prüfling 290 über PE-Karten 325 und 330. Die Digitalkanäle arbeiten in konventioneller Weise: jeder Digitalkanal ist programmierbar in konventioneller Weise zum Ansteuern eines Stiftes des Prüflings gemäß einem vorbestimmten Muster und/oder zum Erfassen digitaler Signale an einem Stift des Prüflings für den Vergleich mit einem erwarteten Reaktionsmuster.

Jeder Analogkanal hat ein Mischsignaldigitaluntersystem 335, einen programmierbaren digitalen Signalprozessor (DSP)-Modul 340 und analoge Stiftelektroniken 345 (PE). Das Mischsignaldigitaluntersystem 335 umfaßt Taktsignalgeneratoren 240, einen Analogsequenzermodul 350 und einen Analogmeßsequenzermodul 355. Die Analog-PE 345 umfaßt eine Wellenformquelle (WFS) 380, Eingangs-(I/P)- und Ausgangs-(O/P)-Filter 385 und einen Wellenformmesser 390.

Digitale Repräsentationen von Analogsignalmustern, die an den Prüfling anzulegen sind, werden mittels Analogsequenzermodul unter Steuerung des pro-

grammierbaren DSP-Moduls 340 synchron mit Taktsignalen von Taktgebern 240 erzeugt. Diese digitalen Repräsentationen werden der Wellenformquelle 380 zugeführt, die sie in analoge Stimuliersignale umsetzt. Die analogen Stimuliersignale können über Filter 385 nach Bedarf geführt werden und dem Prüfling 290 über die Lastplatine 320 zugeführt werden.

Analoge Reaktionssignale vom Prüfling 290 gelangen über die Lastplatine 320 zum Wellenformmesser 390 zwecks Digitalisierung. Die analogen Reaktionssignale können über Filter 385 nach Bedarf geführt werden. Digitale Repräsentationen der analogen Reaktionssignale gelangen zum Analogmeßsequenzermodule 355 für Speicherung und für Nachbearbeitung und Analyse durch den DSP-Modul 340.

Fig. 4 zeigt ein detaillierteres Blockdiagramm von Signalpfaden des Testsystems der Fig. 3, bei der gleiche Bezugszeichen gleiche Komponenten bezeichnen. In der dargestellten Ausführungsform umfaßt jeder Analogkanal Analogstiftelektronikkarten (PE) (Wellenformquelle 380, Filtermodul 385 und Wellenformmesser 390), zwei Analogsequenzerkarten (Analogquellensequenz 350 und Analogmeßsequenz 355) und einen DSP-Modul 340. Jeder Kanal kann für eine bestimmte Analogsignalbandbreite und Auflösung optimiert sein. Die Analog-PE-Karten befinden sich vorzugsweise in einem rauscharmen Testkopf 400 mit den digitalen PE-Karten.

Für jeden Kanal ist ein Analogquellensequenzermodule 350, ein Analogmeßsequenzermodule 355 und ein DSP-Prozessormodule vorgesehen. Diese Module befinden sich vorzugsweise in einem Systemhauptrechner oder Analogkäfig 402. Der Analogquellensequenzermodule 350 umfaßt einen first-in-first-out-Speicher 470, einen Sequenzer 472, einen Synchronisierer 474, einen Sequenzerspeicher 476 und einen Multiplexer 478. Der Analogmeßsequenzermodule 355 umfaßt einen Multiplexer 480, einen Datenpacker 482, einen Datenformatierer 484 und einen Synchronisierer 486. Der Analogquellensequenzermodule 350 und der Analogmeßsequenzermodule 355 werden Taktsignale ausgewählter Frequenz von einem für jeden Kanal vorgesehenen Analogtaktsignalgenerator 240 synchronisiert, der Taktsignale in einem Bereich von beispielsweise 125 MHz bis 250 MHz erzeugen kann. Der auswählbare Takt ermöglicht den Analogkanälen, Signale dem Prüfling zuzuführen und Signale von dem Prüfling zu messen bei einer gewünschten Frequenz oder Datenrate. Beispielsweise kann es erwünscht sein, ein Modem durch Kommunikation mit der Komponente bei einer spezifizierten international genormten Frequenz ITU-TSS (früher CCITT) zu testen.

Der Analogquellensequenzermodule 350 sorgt für die Laufzeiterzeugung von komplexen Wellenformen, die dem Prüfling über die Wellenformquelle 380 und optional über ein Filter 385 zugeführt werden. Der Analogquellensequenzermodule 350 kann auch digitale Repräsentationen von Analogsignalen durch einen Subroutinespeicher und Datenalternierungsmultiplexer 405 an Digitalstiftscheiben 410 liefern. Die digitalen Repräsentationen werden verwendet, um Eingangsstifte des Prüflings 290 durch die Digitalstiftelektroniken 415 anzusteuern. In einer Ausführungsform sind 56 Digital-PE-Einheiten 415 in dem System vorgesehen und das vom Benutzer definierte Testprogramm kann sie auswählen je nach Bedarf für die Verwendung durch die Analogkanäle.

Daten von digitalen PE-Karten 420 gelangen durch

Digitalstiftscheiben 425, einen Multiplexer 430 und einen Analogmultiplexer 435 zum Analogmeßsequenzer 355. Der Analogmeßsequenzermodule 355 transferiert Daten von digitalen PE-Karten 420 oder Wellenformmeßkarten 390 in den DSP-Modul 340. Analogmeßsequenzermodule 355 formatiert die Daten und transferiert sie in den DSP-Modul 340.

Jeder DSP-Modul 340 umfaßt eine DSP-Hardware 440 und einen Überordner 445. Von AMS 355 einlaufende Daten gelangen durch einen ECL-TTL-Umsetzer 450 und dann über einen Schalter 455 an einen der Speicher 460 oder 465. Daten, geliefert von der DSP-Hardware 440 zum Überordner 445 gelangen durch einen Zwischenspeicher 466 und TTL-ECL-Umsetzer 468 an den Analogquellensequenz 350.

Die DSP-Hardware 440 kann eine kommerziell erhältliche Prozessorplatine sein, wie die Signalverarbeitungsplatine, die von Ixthos, Inc. in Silver Spring, Maryland, unter der Modellbezeichnung IxD7232 verkauft wird. Fig. 5 zeigt die Architekturübersicht der IxD7232-Platine. Jeder von einem Paar von digitalen Signalprozessoren 500, 505 kommuniziert über einen entsprechenden Datenbus 510, 515 mit dem Überwacher 445 mit einem zugeordneten Datenspeicher 520, 525 mit einem Inter-DSP-first-in-first-out-Speicher (FI-FO) 530 und mit einem Globalspeicherüberwacher 535. Jeder der digitalen Signalprozessoren 500, 505 kommuniziert über einen entsprechenden Programmbus 540, 545 mit einem zugeordneten Programmspeicher 550, 555. Die Programmbusse 540, 545 ermöglichen auch die Kommunikation durch einen Platinensteuerer (VME-Schnittstelle 560 zu einem VME-Bus 565 und seriellen Ein-/Ausgängen 570, 575).

Datenspeicherbänke 520, 525 und der Globalspeicher 535 werden in den entsprechenden DSP-Prozessoradreibraum kartiert, wodurch dem DSP-Prozessor ermöglicht wird, auf diese Elemente wie gewöhnliche Speicher zuzugreifen. Adreßgeneratoren (nicht dargestellt) sind für den DSP-Prozessor zugreifbar, jedoch nicht für den Analogmeßsequenzer 355.

DSP-A 500 setzt den Adreßgenerator A auf eine Stelle in dem Adreibraum des Überwacherspeichers 460 oder 465. Während der DSP-A 500 aus einer Speicherbank (beispielsweise Speicher 460) ausliest, kann AMS 355 Daten in der anderen Speicherbank speichern (beispielsweise Speicher 465). DSP-A 500 schaltet dann die Bänke um und während AMS 355 Daten in die erste Bank (beispielsweise Speicher 460) einschreibt, kann DSP-A 500 Daten aus der zweiten Bank (beispielsweise Speicher 465) auslesen. Wenn in der Speicherbank genug Platz für alle Eingangsvorgänge vorhanden ist, kann DSP-A 500 den Adreßgenerator A so setzen, daß Mehrfacheingangsvorgänge in eine Bank gezwungen werden, ohne die Befürchtung, daß der hochgeschwinde AMS 445 einen vorherigen Eingang überschreibt. Wenn ein Eingang in einer Speicherbank erhalten wird, kann DSP den AMS aussperren, um ihn daran zu hindern, Daten in den Überwacher zu schreiben.

Die Eingangs/Ausgangs-(I/O)-Methodologie für AMS 355 kann für zwei unterschiedliche Moden gesteuert werden: Block bzw. Realzeit. Im Blockmodus wird nur DSP-A 500 für die Verarbeitung von Daten vom AMS 355 verwendet. In diesem Modus werden alle Bänke des Überwachers 445 von DSP-A 500 gesteuert. Die Größe der Register (nicht dargestellt) im DSP-A 500 entsprechend der Eingangsdatenzählung ist gleich der gesamten Speichergröße des Überwachers 445. Der Überwacher 445 hat einen Zähler (nicht dargestellt) ent-

sprechend der transferierten Datengröße. DSP-A 500 kennt die Basisadresse im Überwacher 445, von wo die einlaufenden Daten beginnen. Auch frischt der DSP-A 500 einen Zeiger zu der Adreßstelle auf, wo die laufenden Daten abgespeichert werden. DSP-A kann die Datenadreßerzeugung im Überwacher 445 entsperren und sperren, wodurch dieser von dem AMS abgetrennt wird. DSP-A 500 kann auch den Überwacher rücksetzen.

Der Überwacher 445 hat die Fähigkeit, einen Interrupt zum DSP-A 500 am Ende des Eingangsblocks von Daten zu senden. Der Überwacher 445 hat die Fähigkeit, einen Interrupt für DSP-A 500 nach einer spezifizierten Anzahl von Eingangsblöcken von Daten zu erzeugen. Der Überwacher 445 kann zwischen den Speicherbänken 460, 465 umschalten, wenn eine Bank voll ist, ohne irgendeinen Verlust von Daten. Der Überwacher kann so aufgebaut werden, daß er zu der nächsten Bank umschaltet nach Empfang einer bestimmten Blockgröße.

Blockmodus. Basierend auf der Größe des Einfangs für den laufenden Test setzt DSP-A 500 einen Transferzähler im Überwacher 445 entsprechend den Daten, die in dem Testprogramm spezifiziert sind. Wenn der Überwacher 445 einen Datenblock empfängt (auch als eine Aufzeichnung bekannt) von AMS 355 entsprechend der im Transferzähler spezifizierten Größe, erzeugt der Überwacher 445 einen Interrupt für den DSP-A 500. Der DSP-A 500 kann den Überwacher 445 so konfigurieren, daß ein Interrupt nur nach einer bestimmten Anzahl von Eingangsblöcken vom AMS 355 erzeugt wird. Wenn DSP-A 500 den Interrupt vom Überwacher 445 erhält, beginnt er mit der Verarbeitung der Daten in den Speicherbänken. DSP-A 500 kann keinen Zugriff auf die Bank nehmen, in die AMS 355 Daten einschreibt. Zusätzlich kann DSP-A 500 die notwendigen Register für den nächsten Eingang vom AMS 355 aufbauen, während AMS 355 Daten schreibt. In diesem Modus hat DSP-A 500 Zugriff auf alle Banken mit Ausnahme derjenigen, auf die AMS 355 gerade Zugriff hat.

Fig. 6 und 7 zeigen den Überwacher (SM) 445 im Blockmodus des Betriebs. In Fig. 6 schreibt AMS 335 in Speicherbank 460, während DSP-A 500 Daten aus Speicherbank 465 liest. Ein Register 600 im SM 445 fällt einen Wert "Xferlen", der die Größe des Eingangsblocks, beispielsweise 1024 Bytes, repräsentiert, und einen Wert "Xfercount", der den Zählstand der Daten repräsentiert, die vom AMS 335 zur Bank 460 transferiert werden. In Fig. 6 hat DSP-A 500 die Register so vorbereitet, daß sie Daten vom AMS 335 akzeptieren. Während AMS-Eingaben der Daten in Bank 460, kann DSP-A 500 die Daten in der Bank 460 nicht in Zugriff nehmen, doch wird DSP-A 500 die Register für die nächste Eingabe vorbereiten. Am Ende der spezifizierten Transferlänge sendet SM 445 einen Interrupt zu DSP-A 500. DSP-A 500 ändert dann den Zugriff der AMS 335 von Bank 460 auf Bank 465, wie in Fig. 7 gezeigt. Da die notwendigen Register bereits von DSP-A 500 vorbereitet waren, erfolgt keinerlei Datenverlust. An diesem Punkt verarbeitet DSP-A 500 die Daten in Bank 460, während AMS 335 die Bank 465 auffüllt.

Realzeitmodus. Im Realzeitbetriebsmodus werden Daten kontinuierlich von AMS 335 zu DSP-A 500 transferiert.

Ein bevorzugtes Mischsignal-Testsystem gemäß der Erfindung, wie hier beschrieben, basiert auf digitalen Untersystemen und benutzt solche des Schlumberger ITS 9000FX Digital-Testsystems. Jenes Digital-Testsystem ist beispielsweise beschrieben in Schlumberger

ITS 9000FX Hardware Reference Manual, Veröffentlichungsnummer 57010045, Revision 4, ECO 17313, August 1993, veröffentlicht von Schlumberger Technologies Inc., San Jose, Kalifornien; auf den Inhalt dieser Veröffentlichung wird verwiesen. Das ITS 9000FX-System umfaßt eine Software-Umgebung, die bekannt ist als "ASAP" (für Advanced Symbolic ATE Programming = fortschrittliche symbolische ATE-Programmierung), welche das Einrichten und Programmieren von Tests vereinfacht. Zum Erfüllen der Mischsignal-Testerfordernisse sind Leistungs- und Masseverteilungen niedrigen Rauschens vorgesehen, ein analoges Subsystem und Instrumentierung werden hinzugefügt, und die ASAP-Software-Umgebung wird erweitert, um Werkzeuge für die Steuerung von Mischsignal-Test-Hardware bereitzustellen.

Fig. 8 ist ein Blockdiagramm zur Darstellung der Gesamtarchitektur eines Mischsignal-Testsystems in einer Ausführungsform gemäß der Erfindung, die auf dem ITS 9000FX-System basiert. Der DSP-Modul 440 ist in einem CPU-Käfig untergebracht und kommuniziert über VME-Bus 565 mit einer zentralen Verarbeitungseinheit (wie einer "Force"-CPU, basierend auf einem Sparc-Prozessor) 805 und Speicher 810. CPU 805 kommuniziert auch mit einer Arbeitsstation 815 mit einem Bildschirm 820 und Eingabe/ Ausgabe-Komponenten (nicht dargestellt) für die Programmierung und Steuerung des Testsystems durch den Benutzer. CPU 805 kommuniziert auch mit einem Systemzustandssteuerer (SSC) 825. Eine Schnittstelle zwischen VME und Tester (VTI) 828 ermöglicht die Kommunikation über einen Hochgeschwindigkeitsbus zwischen Elementen, angebracht am VME-Bus 565 und Hochgeschwindigkeitsschnittstellenmodulen (HSI), die sich in anderen Kartenkäfigen befinden, beispielsweise über HSI 830 im Steuerkäfig (C) 835, HSI 840 im Hochgeschwindigkeitstftkäf (H) 845 (und über solche HSI-Einheiten in den anderen sechs H-Käfigen), und über HSI 850 in dem Analogkäfig (AN) 855. Die C-Käfig-Schnittstellen an alle Hochgeschwindigkeitstfttscheibenkarten für die Gesamtzeitlage- und Adreßerzeugung. Zusätzlich zu dem HSI ist jeder H-Käfig mit einer Taktpufferkarte und bis zu 16 Stiftscheibenkarten ausgestattet. Jede Stiftscheibenkarte steuert vier Testkopfskanäle. Zwei Subroutinespeicher und alternative Datenmultiplexmodule (SMADM) pro H-Käfig können angebracht sein, um jeder 32 Kanäle zu steuern.

VTI 828 ermöglicht auch die Kommunikation über einen Testkopfbus zwischen Elementen, zugeordnet dem VME-Bus 565 und einem Testkopfschnittstellenmodul (THI) 858 in dem Testkopf, beispielsweise für die Kommunikation von Einrichtungsinformationen von CPU 805 an die verschiedenen Elemente des Analogkanals vor dem Test und für den Gewinn von Information nach dem Test. Zeitmeßeinheit 250 liefert ein digitales Taktsignal bei einer ausgewählten Frequenz an einen Testperiodengenerator-(TPG)-Schaltkreis 860. Ein Hauptsequenzsteuerspeicher (MSCM) 865, ein Befehlsdecoder 870, ein Entstörsteuerer 875 und ein Taktpuffer 880 sind ebenfalls in dem Steuerkäfig (C) 835 enthalten. Taktpuffer 885 bzw. 890 sind ebenfalls in dem H-Käfig 845 bzw. AN-Käfig 855 enthalten. Steuerbarer Subroutinespeicher und alternativer Datenmultiplexer 405 können digitale Muster von dem Analogquellensequenz 350 oder anderen optionalen Quellen auswählen und speichern (SCAN und Automatik-Programmengenerator-(APG)-Musterquellen). Ein Gleichstromuntersystem (DC) 895 ist für Messung von Gleichspannungscharak-

teristiken des Prüflings vorgesehen.

Fig. 9 ist ein Flußdiagramm eines Planerprogramms, das in dem Testerprozessor (beispielsweise CPU 805) läuft für das Steuern des übergeordneten Betriebs eines solchen Testsystems. Der Betrieb beginnt mit Block 900. Im Block 910 überprüft das Programm, ob der Tester für Mischsignal-Testen ausgerüstet ist. Falls nicht, nimmt der Planer einen Digitaltest als durchzuführen an und geht bei Schritt 915 dazu über, die ITS 9000FX-Digitaltester-"ASAP"-Software-Umgebung für einen digitalen Test zu verwenden. Falls ja, bestimmt der Planer bei Schritt 920, ob irgendwelche Tests auf ihre Ausführung warten. Falls nein, beendet der Planer den Betrieb bei Schritt 925. Falls ja, bestimmt der Planer bei Schritt 930, ob ein Mischsignal-Test auszuführen ist. Falls nein, geht der Planer bei Schritt 935 weiter zum Ausführen eines digitalen Tests unter Verwendung der "ASAP"-Software-Umgebung. Falls ja, überprüft der Planer bei Schritt 940, ob der auszuführende Test von einem Software-Werkzeug zu steuern ist, das die Bezeichnung "LoadboardTool" trägt. Wenn der auszuführende Test von dem LoadboardTool zu steuern ist, wartet der Planer bei Schritt 945 auf die Beendigung aller Mischsignal-Tests und überprüft dann bei Schritt 950, ob der DSP Einfangdaten für den laufenden Test halten kann. Falls nein, bleibt der Planer im Leerlauf bei Schritt 955, bis der DSP eine Speicherbank freigibt. Falls ja, instruiert der Planer den DSP bei Schritt 960, die laufenden Einfangdaten zu halten. Der Planer überprüft dann bei Schritt 965, ob der laufende Test von der LoadboardTool-Software gesteuert wird. Falls nein, geht der Fluß weiter zu Schritt 920 (Markierung "A"). Falls ja, wartet der Planer bei Schritt 970, daß der laufende Mischsignal-Test beendet wird und geht dann zu Schritt 920 über (Markierung "A"). Wenn keine weiteren Tests auszuführen bleiben, beendet der Planer den Betrieb bei Schritt 925.

LoadboardTool ist eine Software-Schnittstelle, die einfachen Benutzerzugriff zu den funktionellen und physikalischen analogen Fähigkeiten des Testers bietet. Sie sorgt für die Ausführung des Mischsignal-Tests, Überwachung des Testzustands und Steuerung der in dem Test involvierten Instrumente. Fig. 10 ist eine funktionale Wiedergabe dessen, was der Benutzer auf dem Bildschirm 820 sehen würde, wobei eine Repräsentation eines Prüflings 1000 und seiner Stifte gezeigt ist. Ferner sind Repräsentationen von Instrumenten gezeigt, die der Benutzer funktionell mit Stiften des Prüflings verbunden hat durch Manipulation des Bildschirms mit einer Zeiger/Auswahl-Einrichtung wie einer Maus. Beispielsweise sind eine Wellenformquelle WF1 und eine Wellenformmessung WM1 mit Prüflingsstiften 1 und 2 verbunden; eine Digitalmusterquelle DP1 ist mit Prüflingsstift 3 verbunden; eine andere Digitalmusterquelle DP2 ist mit Prüflingsstift 4 verbunden, und so weiter. Der Bildschirm zeigt den laufenden Zustand der Lastplatinen-Relais, der bestimmt wird durch den Zustand der Relais-Steuerung ("C"-Bits) und anderer Testparameter, die der Benutzer vorgibt. Wenn einmal die Vorgabe beendet ist, nimmt die LoadboardTool-Software andere ASAP-Werkzeuge zu Hilfe, wie sie benötigt werden, um den Test auszuführen (beispielsweise Werkzeuge für das Setzen von Mustern, Zeitlagen, Pegeln, Gleichspannungswerten und dergleichen).

Über LoadboardTool kann der Benutzer auch spezialisierte Software-Werkzeuge einbinden für die Berechnung für Signalquelle und -messung, Cbit-Steuerung, Analogtaktsteuerung, Wellenformerzeugung und -messung, Referenz, Produktivität, Spezifikationszusammen-

fassung, Systemzustand und Sequenzen. Jedes der Werkzeuge, das spezifische Hardware steuert, bringt vorzugsweise ein Blockdiagramm der Hardware zur Wiedergabe und zeigt an, welche Teile des Blockdiagramms durch den Benutzer verändert werden können.

Ein "DSPTool" ermöglicht dem Benutzer, die DSPs zu programmieren für die Ausführung der entsprechenden Transferfunktionen. Eine Graphikchnittstelle ist vorgesehen, wie beispielsweise in Fig. 11 gezeigt, zusammen mit Standard-Büchereifunktionen und benutzerkodierten Funktionen für das Programmieren der DSPs. Dieses Werkzeug verleiht dem Benutzer die Möglichkeit, in geschlossener Schleife ablaufende Tests zu schreiben und bei der Entstörung von Programmen die Möglichkeit, Abbruchpunkte bei Funktionen zu setzen und Matrixdaten wiederzugeben. Fig. 11 zeigt bei 1100 eine Probenschirmwiedergabe des DSPTool. Ein Menü von Aktenfunktionen ist bei 1105 gezeigt, ein Menü von Edierfunktionen ist bei 1110 gezeigt, ein Menü von Operationsfunktionen ist bei 1115 gezeigt, ein Menü von Rechnungsfunktionen ist bei 1120 gezeigt, und eine vom Benutzer konfigurierbare Werkzeugbar ist bei 1125 gezeigt. Ein Menü von Entstörfunktionen ist bei 1130 gezeigt. Jeder der Menügegenstände kann von dem Benutzer mit Hilfe einer Maus oder eines Tastenfeldes ausgewählt werden, um gewünschte Systemfähigkeiten abzurufen.

Fig. 12 zeigt die Menü-Hierarchie von Operationsfunktionen einschließlich eines breiten Bereichs von Standard-Büchereifunktionen, die als Vektor, Skalar und DSP kategorisiert sind, wie auch benutzerdefinierte Funktionen. Die DSP-Funktionen umschließen Algorithmen für bekannte Operationen, wie Berechnen eines Hanning-Fensters, Ausführen einer komplexen schnellen Fourier-Transformation (FFT) und Berechnen einer Zeitdomänen-Autokorrelation.

Ein Muster einer vom Benutzer definierten Sequenz von Operationen ist in Fig. 11 wiedergegeben, in der das Signal von einem A/D-Umsetzer durch ein Tiefpaßfilter (LPF) zu führen ist, wobei ein Signal von einer abgespeicherten Akte einer Hanning-Fensteroperation zu unterwerfen ist. Die Resultate dieser beiden Operationen werden jeweils einer schnellen Fourier-Transformation (FFT) unterworfen, gefaltet (CONV) und dann einer inversen schnellen Fourier-Transformation (IFFT) unterworfen zum Erzeugen des gewünschten Ergebnisses. Durch Aufbauen eines Diagramms wie in Fig. 11 kann der Benutzer eine gewünschte Sequenz von Operationen ohne detaillierte Kenntnis des Systembetriebs aufbauen. Die ASAP-Software-Umgebung erzeugt ein detailliertes Testprogramm von der vom Benutzer erzeugten graphischen Wiedergabe des Testablaufs.

Ein "MeasureTool" ermöglicht dem Benutzer, die Meßinstrumentierung zu programmieren. MeasureTool ermöglicht die Auswahl eines von mehreren Moden: HAWM (high-accuracy waveform measure = hochgenaue Wellenformmessung), HFWM (high-frequency waveform measure = Hochfrequenz-Wellenformmessung) oder Digitalstift. In jedem Modus werden das entsprechende Schaltungsdiagramm und das Meßsequenzdiagramm wiedergegeben. Die Wiedergabe umfaßt Blöcke, die den Benutzer aufmerksam machen, die Parameter einzustellen. Ein Beispiel eines MeasureTool-HAWM-Diagramms ist in Fig. 13 gezeigt. Die Hauptfunktionalität des MeasureTool im HAWM-Modus besteht darin, das Audiofilter zu setzen, die entsprechenden hochgenauen Meßoptionen zu setzen und eine Option bereitzuhalten für die Messung der Spannungsaus-

gänge der Wellenformmeßinstrumentierung unter Verwendung des Multimeters. Die Hauptfunktionalität des MeasureTool im HFWM-Modus besteht darin, die entsprechenden Hochfrequenzmeßoptionen zu setzen, das Videofilter zu setzen und eine Option für die Messung der Spannungsausgänge der Wellenformmeßinstrumentierung unter Verwendung des Multimeters bereitzustellen. Die Hauptfunktionalität des MeasureTool im Digitalstiftmodus ist das Kartographieren der Digitalstifte in dem Meßsequenzer durch Steuerung des HCA-GE-Bitkartographiermultiplexers und des Analogmeßmultiplexers. Die Analogmeßsequenzer-Hardware wird ebenfalls mit dem MeasureTool gesteuert. Die Hauptfunktionalität der Analogmeßsequenzerchnittstelle besteht darin, dem Benutzer zu ermöglichen, den Datenlauf zu wählen entweder direkt zum DSP oder durch den Analogmeßsequenzer zum DSP, und eine Schnittstelle für die folgenden Parameter bereitzustellen: Wellenformabschätzblock, Taktquelle, Taktfrequenz, Start- und Stop-Trigger, Datenformate und Dateneinfangmodus.

Ein "SourceTool" bietet dem Benutzer Techniken, dem Prüfling die analoge Wellenform/abgetastete Daten, zuzuführen. Abhängig von den Einstellungen der Hardware im LoadboardTool wird das entsprechende Instrumentenschaltungsdiagramm in diesem Werkzeug dargestellt. Das WaveformTool kann von diesem Werkzeug für die Wellenformerzeugung zu Hilfe genommen werden. Das SourceTool arbeitet in einem von drei Moden, entweder HAWS (hochgenaue Wellenformquelle), HFWS (Hochfrequenzwellenformquelle) oder Digitalstift. In jedem Modus werden das entsprechende Schaltungsdiagramm und das Quellensequenzdiagramm angezeigt. Fig. 14 zeigt ein Beispiel einer SourceTool-HAWS-Diagrammwiedergabe. Die Hauptfunktionalität des SourceTool im HFWS-Modus ist das Einstellen der entsprechenden Hochfrequenzquellenoptionen und des Videofilters. Die Hauptfunktionalität des SourceTool im Digitalstiftmodus ist das Kartieren der Digitalstifte in dem Quellensequenzdiagramm. Die Analogquellensequenzdiagramm-Hardware wird ebenfalls mit dem SourceTool gesteuert. Die Hauptfunktionalität der Analogquellensequenzdiagramm-Hardware besteht darin, dem Benutzer die Auswahl zu geben, die Analogquellensequenzdiagramm-Hardware zu umgehen, wenn die Daten von dem DSP dem Prüfling zugeführt werden, und eine Schnittstelle für die folgenden Parameter bereitzustellen: den Wellenformerzeugungsblock, die Taktquelle, die Taktfrequenz und die Start- und Stop-Trigger.

Vorzugsweise sind noch weitere für den Benutzer zugängliche Software-Werkzeuge vorgesehen. Beispielsweise ermöglicht ein Reference-Tool die Benutzersteuerung der Referenzquelle und eine Wiedergabe einer Funktionsrepräsentation des Schaltungsdiagramms. Ein WaveformTool stellt ein Verfahren zum Erzeugen graphischer Wiedergaben von Wellenformen, die zu erzeugen sind, bereit. Ein CbitsTool steuert die "C"-bit-Relais unter Wiedergabe eines Blockdiagramms aller Lastplatinen-Benutzer-Relais-Hardware und Anzeige, welche Teile durch den Benutzer austauschbar sind. Ein AnalogClockTool unterstützt die Manipulation des Benutzers der Analogtaktgabe und der Phasenrauschquell-Hardware des Testers unter Wiedergabe von Diagrammen der Analogtakt- und Phasenrauschquell-Hardware, Steuerung des Haupttaktgebers und Anzeige, welche Teile des Diagramms durch den Benutzer veränderbar sind. Ein SequenceTool ermöglicht dem Benutzer, die Reihenfolge von Testeraktivitäten zu spe-

zifizieren, die während eines Mischsignal-Tests auszuführen sind.

Die beschriebenen graphischen Software-Werkzeuge werden vorzugsweise verwendet, um die Einrichtung des Systems durch den Benutzer zu vereinfachen, obwohl das Einrichten genauso gut durch andere bekannte Mittel erfolgen könnte, wie direkte Erzeugung eines Testprogramms durch den Benutzer. Wenn einmal die Einrichtung durch den Benutzer definiert ist, überträgt CPU 805 die Einricht- und Steuer- und Sequenzierinformation zu den System-Hardware-Modulen über VTI 828, die HSI-Einheiten in dem C-Käfig, H-Käfigen und AN-Käfig und über THI 858. Die Verfügbarkeit von mehreren Analogkanälen mit Quelle- und Meßinstrumentierung, die flexibel auf diese Weise konfiguriert werden können, ermöglicht dem System, effizient einen weiten Bereich von Mischsignal-Tests auszuführen, von welchen Beispiele unter Bezugnahme auf Fig. 30 bis 32 beschrieben werden.

Fig. 15 zeigt eine graphische Wiedergabe eines ControlTool-Software-Werkzeuges, das brauchbar ist bei der Steuerung der Testeinrichtung und des Testbetriebs. Die Wiedergabe erscheint in einem "Fenster" eines Bildschirms der Arbeitsstation 200 beispielsweise und umfaßt einen Block für die Benutzereingabe eines Testprogrammnamens sowie "Tasten", die gewählt werden können durch Verwendung einer Steuereinrichtung (beispielsweise einer Maus oder einer anderen Zeige- und Antasteinrichtung) zum Aktivieren der benannten Funktionen. Aufbau, Laden, Installieren, Initialisieren, Beginnen, Rücksetzen, Flowtool, Source-Tool, Zeitlage.

Fig. 16 zeigt ein Unterfenster der ControlTool-Wiedergabe, worin Piktogramme gezeigt sind, welche Testsystem T1, T2/M und T3 repräsentieren, verbunden mit der Arbeitsstation 200 und durch diese steuerbar. Tester T1 und T3 sind in diesem Beispiel Digitaltester (beispielsweise Standard-ITS-9000FX-Testsysteme), während Tester T2/M ein Mischsignaltester wie hier beschrieben ist. Die Benutzerauswahl des "T2/M"-Piktogramms bewirkt, daß eine weitere Darstellung wie in Fig. 17 erscheint, worin ein Block, der mit "T2/M" markiert ist, wiedergegeben ist in Verbindung mit Indikatoren, welche entsprechende Testköpfe des Mischsignal-Testers repräsentieren. In dem gezeigten Beispiel indiziert der Indikator für Testkopf Nr. 1 "TH1", daß der Testkopf für die Verwendung verfügbar ist, während der Indikator für Testkopf Nr. 2 zeigt, daß der Testkopf gegenwärtig nicht verfügbar ist. Durch Auswählen des mit "TH1" markierten Indikators befiehlt der Benutzer dem System, sich für Operationen mit Testkopf Nr. 1 vorzubereiten. Wenn der Benutzer einen Testprogrammnamen eingibt und das "Load"-Piktogramm (siehe Fig. 15) auswählt, wird ein Testprogramm I/O-Fenster, wie in Fig. 18 wiedergegeben, zur Anzeige des Zustands der Testprogrammoperationen, beispielsweise "Testprogramm laden", "Testprogramm ist geladen", und so weiter.

Durch Anwählen des "FlowTool"-Piktogramms der Fig. 15 kann der Benutzer ein "FlowTool"-Wiedergabefenster, wie in Fig. 19 gezeigt, aktivieren. Die FlowTool-Wiedergabe unterstützt den Benutzer bei der Definition des Ablaufs eines Tests unter Verwendung der "ASAP"-Werkzeuge des ITS 9000FX-Testsystems. In dem einfachen Beispiel der Fig. 19 beginnt der Test mit einem Block, markiert "Beginn" und geht weiter zu einem "33 MHz-FTest"-Segment, das funktionelle digitale Tests des Prüflings bei einer 33 MHz-Taktgeschwindigkeit ausführt. Wenn der 33 MHz-Test bestanden wird,

fährt der Testablauf fort mit einem Mischsignalsegment "MTest". Im anderen Falle geht der Testablauf weiter mit einem "20 MHz-FTTest"-Segment und so weiter. Wenn ein einzelnes Testsegment, beispielsweise "MTest"-Segment, auszuführen ist, kann der Benutzer das entsprechende Piktogramm aus Fig. 19 anwählen zum Erhalten einer Unterfensterwiedergabe wie in Fig. 20 mit einer Taste, die anzuwählen ist für die Ausführung nur jenes Segments. Die Ausführung eines einzelnen Segments oder einer ausgewählten Gruppe von Segmenten kann brauchbar sein bei der Fehlerbehebung eines Tests, während der Test entwickelt wird.

Die ASAP-Laufzeitexekutionsumgebung besteht aus einem Testprogrammprozeß und Stützprozessen auf der Tester-CPU. Testdaten-Steuer-Software (TDC) stellt Datentransferschnittstelle bereit und unterstützt die Ereignisnotifikation für den Laufzeitprozeß, der auf der Oberseite des UNIX TCP/IP sockelbasierten Kommunikationsprotokoll implementiert wird. Ein TDC-Prozeß, der auf Testercomputer 210 läuft, umfaßt Feststellungen zum Handhaben analoger Laufzeit-Serviceanforderungen, wie: konfiguriere Analogsubsysteme ("load"), installiere Analog-Hardware ("install"), initialisiere Analog-Hardware ("init"), exekutierte Analogtests ("execute"), manage DSP-Ergebnisse und operiere mit Analogdatenblöcken.

Fig. 21 zeigt ein Zustandsdiagramm für den Laufzeitbetrieb eines Mischsignal-Testers in verschiedenen Abläufen entsprechend der Erfindung. Da der Laufzeit-Exekutionsprozeß auf einem ereignisabhängigen Zustandsmodell basiert, beschreibt eine Zustandsdiagrammtechnik die Laufzeitabläufe. Ein Ablauf ist ein Aggregat von Ereignissen, Zuständen, Aktivitäten und Ergebnissen. Die Laufzeitprozeßsteuerung stellt Funktionen bereit für das Manipulieren der analogen Untersysteme innerhalb einer ASAP-Umgebung. Abschnitte des Ablaufs sind in Unterzustandsdiagrammen der Fig. 23 bis 29 gezeigt, um mehr Einzelheiten eines bestimmten Ablaufs wiederzugeben, die Ereignisse, die die Zustandsänderung bewirken, und die Aktivitäten, die aus einer Zustandsänderung resultieren. Ein ausgezogenes Oval repräsentiert einen Anfangszustand und optionale Bedingungen. Eine gestrichelte Linie unterteilt konkurrierende Zustände.

Der Benutzer gibt Befehle durch Arbeitsstation 200 mittels graphischer Benutzerschnittstellentechniken (GUI), wie oben beschrieben, oder einer Operatorschnittstellenkonsole (OIC), wie direkte Eingabe von Befehlen durch ein Tastenfeld. Ein Ereignis kann auftreten, wenn GUI- oder OIC-Benutzer Laufzeitservices anfordern oder der Exekutionsprozeß ein Ereignis erkennt (wie etwa, daß Datenverfolgung erforderlich ist oder irgendein fataler Systemfehler aufgetreten ist oder Testresultate bereit für weitere Verarbeitung sind und so weiter) während eines Laufzeitexekutionszyklus. Ein Zustand, der durch ein Ereignis etabliert wird, steuert die Laufzeitprozeßaktivitäten. Eine Aktivität in Verbindung mit einem Testprogrammzustand ist die Laufzeitoperation. Ein Ergebnis, das erzeugt wird basierend auf den Laufzeitprozeßaktivitäten, ist die Antwort für eine spezifische Anfrage (ein Ereignis).

Load-Ablauf. Der Laufzeitprozeß empfängt eine TDC-Operationsbotschaft mit Botschaftstyp von TDC_SETUP_CONFIG von der TDC-Fazilität bei 2105, wenn der Benutzer einen Ladeservice anfordert durch (1) Auswählen des "load"-Knopfes eines Control-Tool-Bildschirms oder (2) Tippen eines LOAD-Befehls. Nachdem der Laufzeitprozeß diese Operationsbot-

schaft akzeptiert, wird er den "Load"-Zustand bei 2110 initialisieren. Fig. 23 zeigt ein Unterzustandsdiagramm von Aktivitäten eines "Load"-Zustands. Der Systemkonfigurationsaufbau wird dann ausgeführt, beispielsweise durch Eingeben von Fehleraufbauinformation in eine Konfigurationsakte für den Testkopf/die Testköpfe und DSPs bei 2305.

Install-Ablauf. Der Laufzeitprozeß empfängt eine TDC-Operationsbotschaft mit Botschaftstyp von TDC_INSTALL von der TDC-Fazilität, wenn ein Benutzer einen "install"-Service anfordert durch (1) Anwählen des "install"-Knopfes eines ControlTool, (2) Ausführen eines vordefinierten INSTALL-Segments von einem FlowTool, oder (3) Tippen eines INSTALL-Befehls. Nachdem der Laufzeitprozeß diese Operationsbotschaft akzeptiert, wird er den "Install"-Zustand initialisieren bei 2115. Fig. 24 zeigt ein Unterzustandsdiagramm zur Beschreibung von analogen Aktivitäten eines "Install"-Zustands. Diese umfassen das Rücksetzen und Laden von Hardware-Variablen bei 2405 (beispielsweise Lastplatten-Eichwerte), Initialisieren von Generator-(Quellen)-Hardware bei 2410 (beispielsweise Laden von Mustern in den Musterspeicher), Initialisieren der Einfang-(Meß)-Hardware bei 2415, Initialisieren der DSP bei 2420, Herunterladen eines Programms zu dem DSP bei 2425 und Ausführung der Analogsystemeichung bei 2430.

Init-Ablauf. Der Laufzeitprozeß empfängt eine TDC-Operationsbotschaft mit Botschaftstyp TDC_INIT von der TDC-Fazilität, wenn der Benutzer einen Initialisierungsservice anfordert durch (1) Anwählen des "init"-Knopfes auf einem ControlTool-Bildschirm, (2) Ausführen eines vordefinierten INIT-Segments von einem FlowTool oder (3) Tippen eines INIT-Befehls. Nachdem der Laufzeitprozeß diese Operationsbotschaft akzeptiert, wird er den "Init"-Zustand initialisieren bei 2120. Fig. 25 zeigt ein Unterzustandsdiagramm der Analogaktivitäten auf einem "Init"-Zustand, worin jedes Element der Analogkanal-Hardware in einen bekannten Zustand gebracht und die DSP-Einheiten rückgesetzt sind.

Execute-Ablauf. Der Laufzeitprozeß empfängt eine TDC-Operationsbotschaft mit Botschaftstyp von TDC_EXECUTE von der TDC-Fazilität, wenn der Benutzer einen "execute"-Testservice anfordert durch (1) Anwählen des "execute"-Knopfes eines TestTools, beispielsweise wie in Fig. 20 gezeigt, (2) Exekutieren eines Segments von einem FlowTool, beispielsweise wie in Fig. 19 gezeigt, oder (3) Tippen eines EXECUTE-Befehls. Nachdem der Laufzeitprozeß diese Operationsbotschaft akzeptiert, wird er den "Execute"-Zustand initialisieren bei 2125. Bei 2125 der Fig. 21 ist ein Unterzustandsdiagramm wiedergegeben einschließlich "execute segment" 2130, "start of test" 2135, HW_reset_tester 2140 (schnelle Hardware-Initialisierung), "execute test" 2145, was wiederholt wird, bis alle Tests des Testablaufs ausgeführt sind, und "end of test" 2150, was die Testoperation beendet. Andere Abläufe können nach Erfordernis auch vorgesehen werden, wie bei 2155 angedeutet.

Fig. 26 bis 29 zeigen Unterzustandsdiagramme des "Execute"-Zustands. Fig. 26 zeigt mehr Einzelheiten des "continue"-Modus des "execute test"-Zustands 2145, worin der Tester bei 2605 rückgesetzt wird, der Test aufgebaut wird bei 2610, der Mischsignal-Test ausgeführt wird bei 2615 ("do analog test"), die Digitalstiftelektroniken rückgesetzt werden bei 2620 ("RTL setup_restore_pins") und die Sequenz von Zuständen 2605

bis 2620 weiterläuft, bis alle Tests des Ablaufs ausgeführt sind. Der Testaufbau bei Zustand 2610 umfaßt Unterzustände des Aufbaus digitaler Stiftelektroniken bei 2625 ("RTL setup open pins") und Aufbau der Analogkanäle bei 2630 ("ana_setup_analog"). Fig. 27 zeigt ein Unterzustandsdiagramm des Analogkanal-Aufbauzustands 2630 einschließlich des funktionalen Aufbaus der Testinstrumentierung bei 2705 ("level, timing, pattern... setup"), DSP-Aufbau bei 2710, Sequenzerwellenformaufbau bei 2715 ("load analog sequenzer"), Wellenformquellenaufbau bei 2720 ("src_setup_ws") und Wellenformmeßaufbau bei 2725 ("meas_setup_wm").

Fig. 28 zeigt ein Unterzustandsdiagramm des Analogtestzustands 2615 einschließlich des Starttestzustands bei 2805 und seine Unterzustände, welche den Meßsequenzer bei 2810 starten, den Quellensequenzer bei 2815 starten und den funktionellen Test ("ftest") bei 2820 starten. Wenn Stop-Bedingungen auftreten, stoppt der Test bei Zustand 2825 und das Testergebnis wird transferiert für die Verarbeitung in einem DSP bei Zustand 2830. Fig. 29 zeigt ein Unterzustandsdiagramm des Analogteststops. Der DSP wird bei Zustand 2905 abgefragt um festzustellen, ob er seine Operationen abgeschlossen hat, während ein Auszeitakt bei Zustand 2910 überprüft wird. Wenn der DSP seine Operationen beendet hat oder der Auszeitakt gelaufen ist, wird der Sourcesequenzer bei 2915 gestoppt, der Meßsequenzer wird bei 2920 gestoppt, der ftest wird bei 2925 beendet und der DSP wird bei 2930 gestoppt.

Fig. 22 zeigt die Reihenfolge der Laufzeitzustände zum Abschließen eines vollständigen Laufzeittest-Exekutionszyklus zusammen mit zugeordneten Funktionen auf höherer Ebene. Das Testprogramm wird geladen und installiert, danach wird der Tester initialisiert. Der Abschnitt zwischen den doppelten vertikalen Linien in Fig. 22 (von "Fast Init" durch "EOT Binning") ist der Exekutionsabschnitt, der in einer Mehrzahl von Möglichkeiten ausgeführt werden kann. In der Produktion würde der Exekutionszyklus ausgeführt, wie durch den Testprogrammablauf vorgegeben, beispielsweise einmal für jede zu testende Komponente, und abgelegt bei dem Befehl, den Test zu beginnen. Während des Programmierens und der Fehlerbehebung kann der Benutzer gerade den "execute test"-Abschnitt oder den "execute segment"-Abschnitt oder den "execute & continue segment"-Abschnitt ausführen.

Es ist festzuhalten, daß der "ftest_start"-Zustand 2820 in Reaktion auf einen "EINST"-Trigger (Enable INSTRument) erfolgt, der den Beginn des ftest signalisiert. Dieser Trigger wird dem Analogquellensequenzer 350 zugeführt und dem Analogmeßsequenzer 355, wie in Fig. 4 gezeigt. Der EINST-Trigger ist ein Synchronisationssignal zwischen dem digitalen Untersystem und den analogen Instrumentierungsuntersystemen und kann beispielsweise ein codiertes 8-Bit-Signal sein mit bis zu 256 eindeutigen Triggerereignissen, geliefert vom Befehlsdecoder 870, wie in Fig. 8 gezeigt. Der EINST-Trigger kann beispielsweise verwendet werden zum Initialisieren des Einfangs eines Analogsignals an einem bestimmten Punkt relativ zu dem an dem Prüfling angelegten Digitalmuster. Er kann auch verwendet werden zum Initialisieren der Quelle (Source), so daß die Analogwellenformstimulation, angelegt an einen Prüfling, immer in derselben Phasenlage relativ zu dem Digitalmuster von Test zu Test und von Prüfling zu Prüfling ist. Der EINST-Trigger kann verwendet werden zum Steuern des Quellensequenzers, so daß Wellenforminformation, gespeichert in dem Wellensequenzer, koordiniert wer-

den kann mit der Rahmenbildungsinformation in den digitalen Stiftscheibenelektroniken, beispielsweise Information des Quellensequenzers, wenn er um einen Datenschnitt weitergehen muß, so daß die Daten in dem Rahmen zu der richtigen Zeit fallen. In ähnlicher Weise kann der EINST-Trigger verwendet werden zum Instruieren des Meßsequenzers, wenn dieser Einfangdaten von einem Rahmen gemäß abgespeicherter Rahmenbildungsinformation zu extrahieren hat.

Fig. 30 zeigt die Hauptsignaloperationen eines Typs von Mischsignal-Tests, bei dem das Signalrauschverhältnis der Sende- und Empfangsseite eines Codec zu bestimmen sind. Ein Codec umfaßt einen Digital-Analog-Umsetzer (D/A) und einen Analog-Digital-Umsetzer (A/D) auf einem einzigen Chip gegebenenfalls mit weiteren Schaltungen zusammen. Die linke Spalte der Fig. 30 zeigt die Operationen beim Testen der Senderseite (D/A-Umsetzer) des Codec. Die rechte Spalte der Fig. 30 zeigt die Operationen beim Testen der Empfängerseite (A/D-Umsetzer) des Codec. Eine digitale Repräsentation eines Testsignals, in diesem Falle eine Sinuswelle, wird in der Tester-CPU 805 erzeugt und in den Speicher des Analogquellensequenzers 350 beim Systemaufbau geladen. Es wird angenommen, daß der Tester nach Befehlen des Benutzers aufgebaut worden ist und daß alle notwendigen Aufbauinformationen über VTI 828, THI 858 und den Hochgeschwindigkeitsbus, verbunden mit den HSI-Einheiten 830, 840, 850 und so weiter kommuniziert worden ist.

Wenn der Test beginnt, überträgt der Analogquellensequenzer 350 die digitale Repräsentation über Pfad C1 des Analogkanals zum Subroutinespeicher und alternierenden Datenmultiplexer 405. Es wird festgehalten, daß der Pfad als Pfad C in Fig. 30 angegeben ist, während es tatsächlich einen solchen Pfad für jeden der vier Analogkanäle gibt. Zur Vereinfachung der Beschreibung wird der Pfad C des Analogkanals 1 als Pfad C1 bezeichnet, Pfad C des Analogkanals 2 wird als Pfad C2 bezeichnet, Pfad E des Analogkanals 1 wird als E1 bezeichnet und so weiter. Zur Vereinfachung der folgenden Beschreibungen versteht es sich, daß, wenn Daten beispielsweise über Pfad D1 transferiert werden, sie von dem Analogquellensequenzer 350 des Analogkanals 1 geliefert werden und von der Wellenformquellenelektronik 380 des Analogkanals 1 empfangen werden. DSP-A des Analogkanals 1 wird als DSP-A1 und jene des Analogkanals 2 als DSP-A2 bezeichnet.

Während des Systemaufbaus werden die Digitalstiftscheibenelektroniken 410 mit vom Benutzer definierter Information darüber versehen, wie die Daten in Rahmen zu bringen und zu formatieren sind für den Prüfling einschließlich Signalpegeln und Zeitlage. Während des Tests werden die Daten auf Pfad C1 in die Rahmen unter Steuerung des Subroutinespeichers und des alternierenden Datenmultiplexers 405 eingesetzt und in den Digitalstiftscheibenelektroniken 410 formatiert, wie für den Prüfling angemessen. Das resultierende Digitalsignal wird dem Prüfling über Digitalstiftelektroniken 415 und Pfad E1 zugeführt.

Der Ausgang des Prüflings ist ein analoges Sinuswellensignal, das über Pfad H1 den Wellenformmeßstiftelektroniken WFM 390 zugeführt wird, wo es digitalisiert wird. Die resultierenden Daten werden über Pfad N1 zum Analogmeßsequenzer 355 gesandt, wo sie in das IEEE-Fließkommaformat umgesetzt werden. Die IEEE-Fließkommatdaten werden über Pfad K1, Überwacher 445 und Pfad L1 dem DSP-A1 500 transferiert. DSP-A1 500 führt eine schnelle Fourier-Transformation bezüg-

lich der Einfangdaten aus, gefolgt von einer Berechnung des Signal/Rausch-Verhältnisses (SNR). Nach Berechnen von SNR hält DSP-A1 das Testergebnis, in diesem Beispiel Daten, welche einen einzelnen Fließkomma-SNR-Wert von 93,5 dB repräsentieren. DSP-A1 500 kann auch andere interessierende Beziehungen berechnen, wenn ihm entsprechende Befehle von dem Benutzer während des Aufbaus gegeben werden, wie einen unechten freien dynamischen Bereich, Gesamtharmonischenverzerrung und so weiter. SNR ist das Verhältnis der Leistung in der Grundfrequenz M zu der Leistung aller anderen Spektralkomponenten von 1 bis N, typischerweise unter Ausschluß von Gleichspannung. Gesamtharmonischenverzerrung ist das Verhältnis der Leistung in der Grundfrequenz M zur Leistung in den Harmonischen der Grundfrequenz, beispielsweise das Verhältnis der Leistung bei f0 zur Summe der Leistung bei 2f0, 3f0 und 4f0. Der unechte freie dynamische Bereich ist das Verhältnis der Leistung im Grundsignal zur Leistung in der nächsthöheren Spektralkomponente.

Während der D/A-Umsetzer des Codec im Analogkanal 1 getestet wird, wird gleichzeitig im Analogkanal 2 der A/D-Umsetzer des Codec getestet. Digitaldaten, welche eine Sinuswelle repräsentieren, werden von dem Analogquellensequenz 350 über Pfad D2 der Wellenformquelle 380 zugeführt. Die Wellenformquelle legt entsprechende analoge Sinuswellen auf Pfad F2 an den A/D-Umsetzer des Prüflings an. Der Ausgang des A/D-Umsetzers sind digitale Daten, die durch Pfad G2 und Digitalstiftelektroniken 420 den Digitalstiftscheibenelektroniken 425 zugeführt werden. In den Digitalstiftscheibenelektroniken 425 werden die Digitaldaten mit logischen Schwellen verglichen und zu den entsprechenden Zeiten abgetastet unter Verwendung von Schwellen- und Zeitlageinformation, definiert durch den Benutzer während des Systemaufbaus. Die resultierenden Digitaldaten auf Pfad I2 sind typischerweise bitweise verschlüsselt, weil die Lastplatine vorzugsweise ausgelegt ist, um jedem Prüflingsstift an den nächstgelegenen Testerstift durchzuverbinden, ohne daß Signalpfad-überkreuzung vorliegt. Die bitweise verschlüsselten Daten auf Pfad I2 werden entschlüsselt durch den H-fail-Multiplexer 430 und Analogmultiplexer 435, wobei die Multiplexer entsprechend während des Systemaufbaus konfiguriert worden sind. Die entschlüsselten Daten auf Pfad J2 werden in Analogmeßsequenzer 355 in das IEEE-Fließkommaformat umgesetzt, das bevorzugte Format für DSP-Operationen, und dem DSP-A2 500 über Pfad K2, Überwacher 445 und Pfad L2 zugeführt. DSP-A2 500 nimmt eine schnelle Fourier-Transformation an den Daten vor und berechnet SNR und irgendwelche anderen von dem Benutzer definierten Parameter. Nach dem Berechnen von SNR hält DSP-A2 das Testergebnis, in diesem Beispiel eine einzige Fließkommazahl, welche den SNR-Wert 97,3 dB repräsentiert.

Bei Beendigung seiner Operationen informiert jeder DSP-A1 und DSP-A2 die CPU 805 darüber. Bei Abfrage durch CPU 805 transferieren DSP-A1 und DSP-A2 die SNR-Werte zur CPU 805, die dann die SNR-Werte gegen Testgrenzen testet, die von dem Benutzer während des Systemaufbaus etabliert worden sind. Die CPU 805 bestimmt, ob der Codec innerhalb der Grenzwerte liegt und den Test bestanden hat, oder außerhalb der Grenzwerte liegt und den Test nicht bestanden hat. Wenn der Tester vier Analogkanäle hat, wie in der beschriebenen Ausführungsform, kann die Sende- und Empfangsseite von zwei solcher Codices gleichzeitig getestet werden. In ähnlicher Weise kann jede Kombination von bis zu

vier A/D- und D/A-Umsetzern gleichzeitig getestet werden.

Fig. 31 zeigt die Hauptsignaloperationen eines Modem-Bitfehlertests, manchmal als Modem-Bitfehlerrate-test bezeichnet. Der Prüfling, ein Modem, ist mit einem Analogsignal anzusteuern, und sein Ausgang wird bezüglich Fehler überprüft. Es wird angenommen, daß entsprechende Aufbauinformation den Testsystemelementen vor Beginn des Tests zugeführt worden sind. Während des Aufbaus werden datenrepräsentierende Wellenformsegmente in dem Speicher des Analogquellensequenzers 350 gespeichert, so daß Daten, die eine Wellenform repräsentieren, von dem Analogwellensequenz 350 in Reaktion auf Rahmenzeiger von dem DSP-B 505 erzeugt werden. Während dieses Tests wird der Analogquellensequenz 350 im DSP-Adreßmodus betrieben, so daß, wenn DSP-B1 505 eine vom Benutzer definierte Sequenz von Rahmenzeigern (a,b,b,a, etc.) (Pfad B1) erzeugt, der analoge Quellensequenz Daten auf Pfad D1 bereitstellt, welche die entsprechende Sequenz von Wellenformsegmenten repräsentieren. Die Daten auf Pfad D1 werden mittels Wellenformquelle 380 in ein Analogsignal umgesetzt, das sie an Prüfling auf Pfad F1 anlegt. In dem dargestellten Beispiel wird der Prüfling angesteuert mit einem frequenzverschiebungsverschlüsselten (FSK)-Signal, obwohl dieselbe DSP-Adressiertechnik verwendet werden kann, um phasenschiebungsverschlüsselte Mehrpegel-, quadraturmodulierte oder andere Signale zu erzeugen. Der Prüfling setzt das angelegte Analogsignal in digitale Worte um, die über Pfad G1 und Digitalstiftelektroniken 420 den Digitalstiftscheibenelektroniken 425 zugeführt werden. In diesem Test werden die erwarteten Werte der Digitalworte gespeichert sein worden in den Stiftscheibenelektroniken 425 während des Aufbaus des Tests. Diese erwarteten Werte sind dieselben wie das benutzerdefinierte Muster, das verwendet wurde zum Erzeugen des Analogsignals zum Prüfling. Digitalstiftscheibenelektroniken 425 vergleichen die Digitalworte im Prüfling mit den erwarteten Werten in Realzeit wie in dem üblichen ITS 9000FX-Digitaltester und kann eine Hardware-Fehlererkennungsflagge setzen, wenn ein Fehler erkannt wird. Abhängig davon, wie der Benutzer den Tester aufgebaut hat, wird der Test beendet, wenn die Fehlererkennungsflagge gesetzt ist, oder kann der Test fortgesetzt werden, um zusätzliche Daten für die Fehlerbehebung zu gewinnen. Die Fehlererkennungsflagge und/oder Daten, die zu verwenden sind für die Fehlerbehebung, können der CPU 805 über HSI 840 und VTI 828 bei Abschluß des Tests zugesandt werden. CPU 805 bestimmt aus der Flagge und/oder Daten, ob der Prüfling den Bitfehlertest bestanden oder nicht bestanden hat.

Der Mischsignal-Tester der vorliegenden Erfindung ist auch geeignet für die Ausführung eines Tests bei einer A/D-Umsetzerregelschleife bezüglich der Codeflankenübergangsgenauigkeit. Ein A/D-Umsetzer hat eine unendliche Anzahl möglicher analoger Eingangswerte über einen kontinuierlichen Bereich, doch nur eine diskrete Anzahl von digitalen Ausgangswerten. Um einen A/D-Umsetzer richtig zu charakterisieren ist es erforderlich, jede der Eingangsspannungen zu kennen, die bewirkt, daß der Ausgang von einem Code zu dem nächsten übergeht. Eine Möglichkeit zur Bestimmung der Codeflankenübergangsspannungen ist das Anlegen von Spannungen an den Prüfling und die Überwachung seiner Reaktion.

Fig. 32 zeigt den primären Signalfuß in einem Ana-

logkanal, wenn ein solcher Test ausgeführt wird. Das Ziel besteht darin, die analoge Eingangsspannung z zu finden, die an dem Prüfling angelegt werden muß, um am Ausgang des Prüflings einen gewünschten digitalen Code DATAz auf der positiven Seite des zu messenden Übergangs zu erzeugen. Die Werte HighLimit und LowLimit sind vom Benutzer definierte Grenzen für die Analogwerte, die an den Prüfling anzulegen sind. Der Wert $x(n)$ ist die Analogspannung, die an den Prüfling während einer gegebenen Iteration n von Testprozessen angelegt wird. Werte $z + \epsilon$ und $z - \epsilon$ sind benutzerdefinierte Grenzen eines akzeptablen Fehlerbandes in dem bestimmten Wert von z (d. h. ϵ ist die akzeptable Auflösung für die Messung von z). Die Variable w ist eine Polaritätsflagge (mit einem Wert von $+1$ oder -1), welche anzeigt, ob der nächste Wert von $x(n)$, der an dem Prüfling anzulegen ist, von der vorhergehenden Iteration ausgehend erhöht oder verringert werden sollte, das heißt, ob das Anlegen des laufenden Wertes $x(n)$ an den Prüfling einen Ausgangscode produziert, der oberhalb oder unterhalb des gewünschten Übergangscodes DATAz liegt. Der Wert $y(n)$ ist ein laufender Mittelwert, der Werte von $x(n)$, die während der letzten K -Durchgänge an dem Prüfling angelegt wurden, worin K eine benutzerdefinierte Zahl von Iterationen für den laufenden Mittelwert ist. Es wird angenommen, daß das Testsystem initialisiert worden ist entsprechend Befehlen durch den Benutzer vor Beginn des Tests.

Gemäß Fig. 32 startet der Prozeß mit variabler $x(0)$, gesetzt auf einen Wert, der einen vom Benutzer abgeschätzten Wert z_0 repräsentiert, der etwas oberhalb oder unterhalb des unbekannten Wertes von z liegt, und mit Variablen w und n initialisiert. Wenn beispielsweise die Variable $x(0)$ von dem Benutzer auf einen Wert etwas unterhalb des erwarteten Wertes von z gesetzt ist, wird die Variable w auf 1 gesetzt zur Anzeige dafür, daß z als größer als der Anfangswert von $x(0)$ erwartet wird. Die Variable n wird initialisiert auf 0 zur Anzeige dafür, daß dies die 0te Iteration der Testschleife zu sein hat. DSP-B1 505 liefert Daten, die den Wert $x(1)$ repräsentieren, auf Pfad B1 zum Analogquellensequenz 350. Für diesen Test ist der Quellensequenz 350 im Durchflußmodus, so daß die Daten passieren ohne Modifikation zum Signalpfad D1 und demgemäß der Wellenformquelle 380 zugeführt werden. Die Wellenformquelle 380 setzt die Daten in eine diskrete analoge Spannung $x(1)$ um und legt den Wert an den Prüfling 290 über Pfad F1 an. Der Prüfling setzt die angelegte Analogspannung $x(1)$ in einen formatierten Digitalcode auf Pfad G1 um. Der Digitalcode passiert durch die Digitalstiftelektroniken 420, unformatiert durch die Digitalstiftscheibenelektroniken 425, läuft durch Multiplexer 430 und 435 und wird als serieller Datenstrom an den Analogmeßsequenz 355 angelegt. Der Analogmeßsequenz 355 konvertiert die seriellen Daten in parallele Daten in Datenpacker 482, setzt die parallelen Daten in IEEE-Fließkommaformat in Formatierer 484 um und liefert die Fließkommarepräsentation des Ausgangs des Prüflings auf Pfad K1. Die Fließkommarepräsentation (DATA_n) geht durch den Überwacher 445 und über Pfad L1 zum DSP-A1 500, wo sie mit dem interessierenden Codeübergang DATAz verglichen werden. Wenn der Wert des Code DATA_n für diese Iteration n größer als oder gleich dem Wert des Codeübergangs DATAz ist, wird w auf -1 für die nächste Iteration gesetzt. Wenn der Wert von Code DATA_n kleiner ist als der Codeübergang DATAz, wird w auf $+1$ für die nächste Iteration gesetzt. DSP-A1 500 sendet den Wert von w über Pfad M1 zum

Speicher 535, wo er für DSP-B1 505 zugreifbar ist.

DSP-B1 505 integriert die Werte von $x(n)$ bei jeder Iteration unter Berücksichtigung der Polarität der Variablen w , beispielsweise $x(n) = x(n-1) + A \cdot w$, worin A eine vom Benutzer definierte Variable ist, die beherrscht, wie stark der Wert von $x(n)$ von einer Iteration der Testschleife zur nächsten inkrementiert wird. Die Variable A kann ein konstanter Wert sein, wird jedoch vorzugsweise durch einen geeigneten Algorithmus bestimmt, der anfänglich dem Wert $x(n)$ erlaubt, sich in großen Schritten zu ändern, bis der Wert von z eingeklebt ist, und der progressiv die Schrittgröße herabsetzt um sicherzustellen, daß der Wert von z mit einer für den benutzerakzeptablen Auflösung bestimmt wird. Ein solcher Algorithmus mit variabler Schrittgröße kann die Anzahl von Iterationen verringern, die erforderlich sind, um das Testresultat mit einer gegebenen Auflösung zu erhalten. DSP-B1 505 hält auch einen laufenden Mittelwert $y(n)$ der Analogspannungen $x(n)$, die an den Prüfling während der letzten K -Durchgänge angelegt wurde, beispielsweise

$$y(n) = \left[\frac{1}{K} \right] \sum_{k=0}^{K-1} x(n-k)$$

worin k ein Summationsindex ist.

DSP-B1 inkrementiert dann den Wert von n und verläßt die Testschleife, wenn der laufende Mittelwert sich um weniger als die vom Benutzer definierte Auflösung bewegt hat, d. h. Schleife verlassen, falls $|y(n) - y(n-1)| < \epsilon$. Wenn der laufende Mittelwert sich nicht innerhalb der vom Benutzer definierten Auflösung befindet, sendet DSP-B1 eine digitale Repräsentation des aufgefrischten Wertes von $x(n)$ über Pfad B1 zum Analogquellensequenz 350. Die Ablaufschleife wird wiederholt, bis der DSP-B1 die Testschleife verläßt. Beim Verlassen der Schleife überträgt DSP-B1 den Wert von $y(n)$ zur Tester-CPU 805.

Wenn man die Spannung über der Zeit im Signalpfad F1 über viele Durchgänge durch die Ablaufschleife überwachen würde, bestünde die Tendenz der Zunahme bis zum überschießen des Wertes z , nachfolgendes Abnehmen bis zum Unterschießen des Wertes z , und so weiter. Das heißt, die Spannung auf Signalpfad F1 würde über viele Durchgänge um die Codeübergangsspannung z her zum oszillieren, etwa wie eine Sägezahnwelle von progressiv abnehmender Amplitude. Wenn dieses Spannungssignal sich um z herum stabilisiert innerhalb des Auflösungsgrenzwertes, der vom Benutzer definiert ist, wie indiziert durch einen stabilisierten laufenden Mittelwert $y(n)$, kann der Wert von $y(n)$ als Codeflankenübergangsspannung genommen werden.

Patentansprüche

1. Vorrichtung für das Mischsignal-Testen eines integrierten Schaltkreisprüflings, umfassend:
 - a. einen Haupttaktgeber (250);
 - b. eine Mehrzahl von digitalen Kanälen, von denen jeder Digitalkanal aufweist
 - i. Digitalstiftscheiben (310, 315) für den Empfang von Zeitlagereferenzen von dem Haupttaktgeber (250) und für die Kommunikation mit Digitalstiftelektroniken;
 - ii. Digitalstiftelektroniken (325, 330), die mit den Digitalstiftscheiben kommunizieren.

ren für das Anlegen digitaler Signale an den Prüfling bzw. Empfangen digitaler Signale von dem Prüfling (290); und
c. eine Mehrzahl von Analogkanälen, wobei
jeder Analogkanal aufweist

- i. einen Analogquellensequenzner (350) unter Steuerung eines DSP-Moduls (340) für das Erzeugen von digitalen Repräsentationen von an den Prüfling (290) anzulegenden Analogsignalen;
- ii. Analogstiftelektroniken (345), die auf den Analogquellensequenzner (350) reagieren für das Anlegen von Analogsignalen an den Prüfling (290) und für den Empfang von Analogsignalen von einem Prüfling (290);
- iii. einen Analogmeßsequenzner (355), der auf die Analogstiftelektroniken (350) reagiert für das Aufbereiten von digitalen Repräsentationen von Analogsignalen, die durch den Prüfling (290) erzeugt wurden; und
- iv. einen programmierbaren DSP-Modul (340) für die Verarbeitung der Repräsentationen von Analogsignalen, abgespeichert in dem Analogmeßsequenzner (355), und für die Zufuhr von Steuerinformation zu dem Analogquellensequenzner (350).

2. Die Vorrichtung nach Anspruch 1, bei der der DSP-Modul (340) einen ersten DSP (500), einen zweiten DSP (505) und mindestens einen Speicher (530, 535) umfaßt, auf den der erste DSP (500) und der zweite DSP (505) Zugriff haben.

3. Die Vorrichtung nach Anspruch 2, bei der der erste DSP (500) programmiert ist zum Empfang von digitalen Repräsentationen von Analogsignalen, vorbereitet durch den Analogmeßsequenzner (355) für die Verarbeitung der digitalen Repräsentationen von Analogsignalen zum Erzeugen eines Resultats, und für das Speichern des Resultats in dem Speicher, und bei der der zweite DSP (505) programmiert ist für den Zugriff auf das abgespeicherte Resultat in dem Speicher und für das Steuern des Analogquellensequenzners in Abhängigkeit von dem Resultat.

4. Die Vorrichtung nach Anspruch 2, bei der der mindestens eine Speicher einen Globalspeicher (535) umfaßt.

5. Die Vorrichtung nach Anspruch 2, bei der der mindestens eine Speicher einen Inter-DSP-first-in-first-out-Speicher (530) umfaßt.

6. Die Vorrichtung nach Anspruch 2, bei der ein DSP-Modul (340) ferner einen Datenspeicher (520) und einen Programmspeicher (550) in Kommunikation mit dem ersten DSP (500) umfaßt, und einen Datenspeicher (525) und einen Programmspeicher (555) in Kommunikation mit dem zweiten DSP (505).

7. Die Vorrichtung nach Anspruch 1, bei der der DSP-Modul (340) eine DSP-Hardware (440) umfaßt, einen ersten Speicher (460) umfaßt, einen zweiten Speicher (465) umfaßt und einen Mehrstellungsschalter (455) umfaßt, der dazu dient, wenn in einer ersten Schaltposition selektiv Einfangdaten von dem Analogmeßsequenzner (355) für die Speicherung im ersten Speicher (460) passieren zu lassen, während Daten in dem zweiten Speicher (465) zugreifbar für die DSP-Hardware (440) sind, und

der Schalter (455) in einer zweiten Schaltstellung dazu dient, selektiv Einfangdaten von dem Analogmeßsequenzner (355) für Speicherung in dem zweiten Speicher (465) passieren zu lassen, während Daten in dem ersten Speicher (460) für die DSP-Hardware (440) zugreifbar sind.

8. Die Vorrichtung nach Anspruch 7, bei der die DSP-Hardware (440) einen ersten DSP (500), einen zweiten DSP (505) und mindestens einen Speicher (530, 535) umfaßt, der für den ersten DSP (500) und den zweiten DSP (505) zugreifbar ist.

Hierzu 25 Seite(n) Zeichnungen

Fig. 1

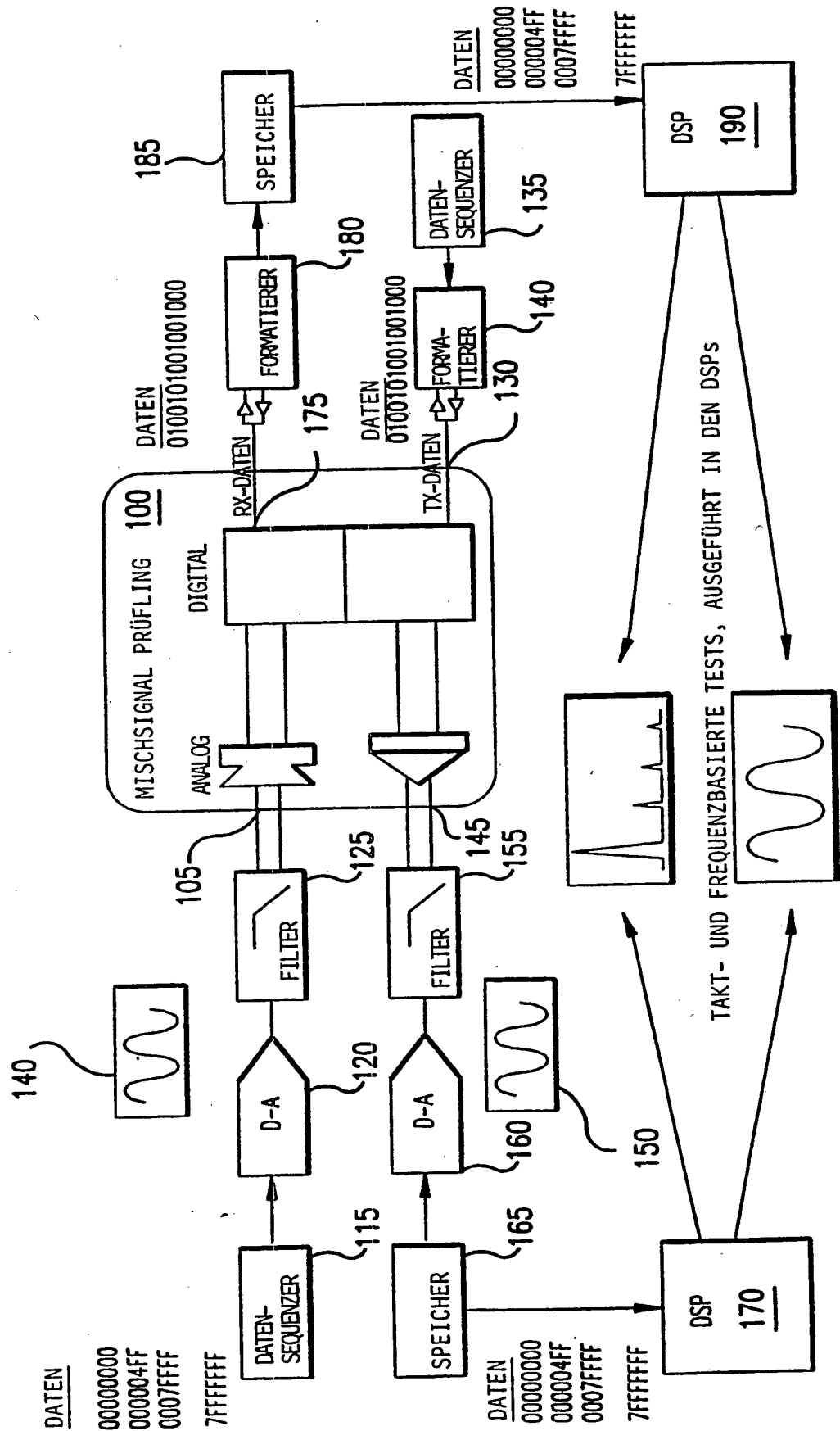
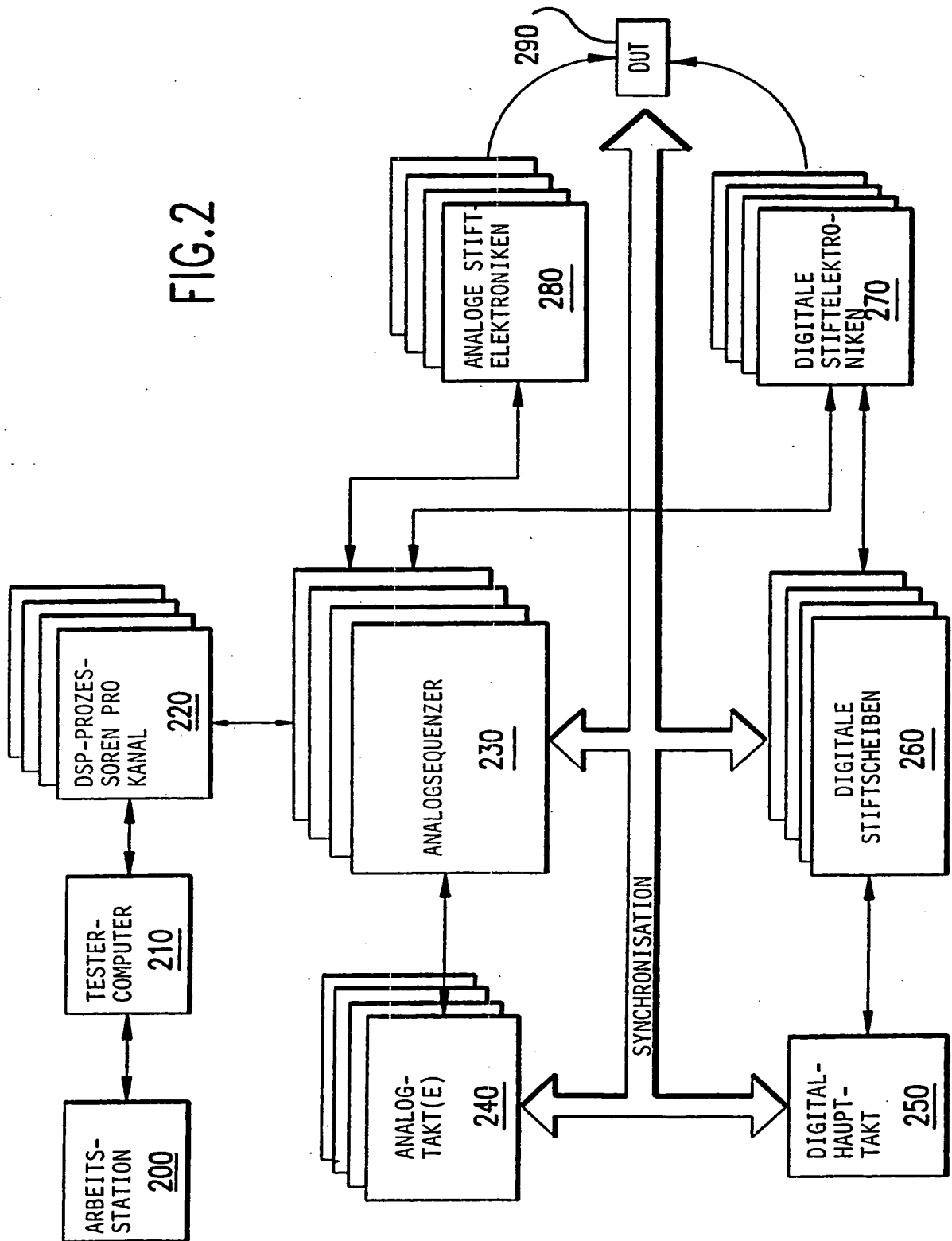


FIG. 2



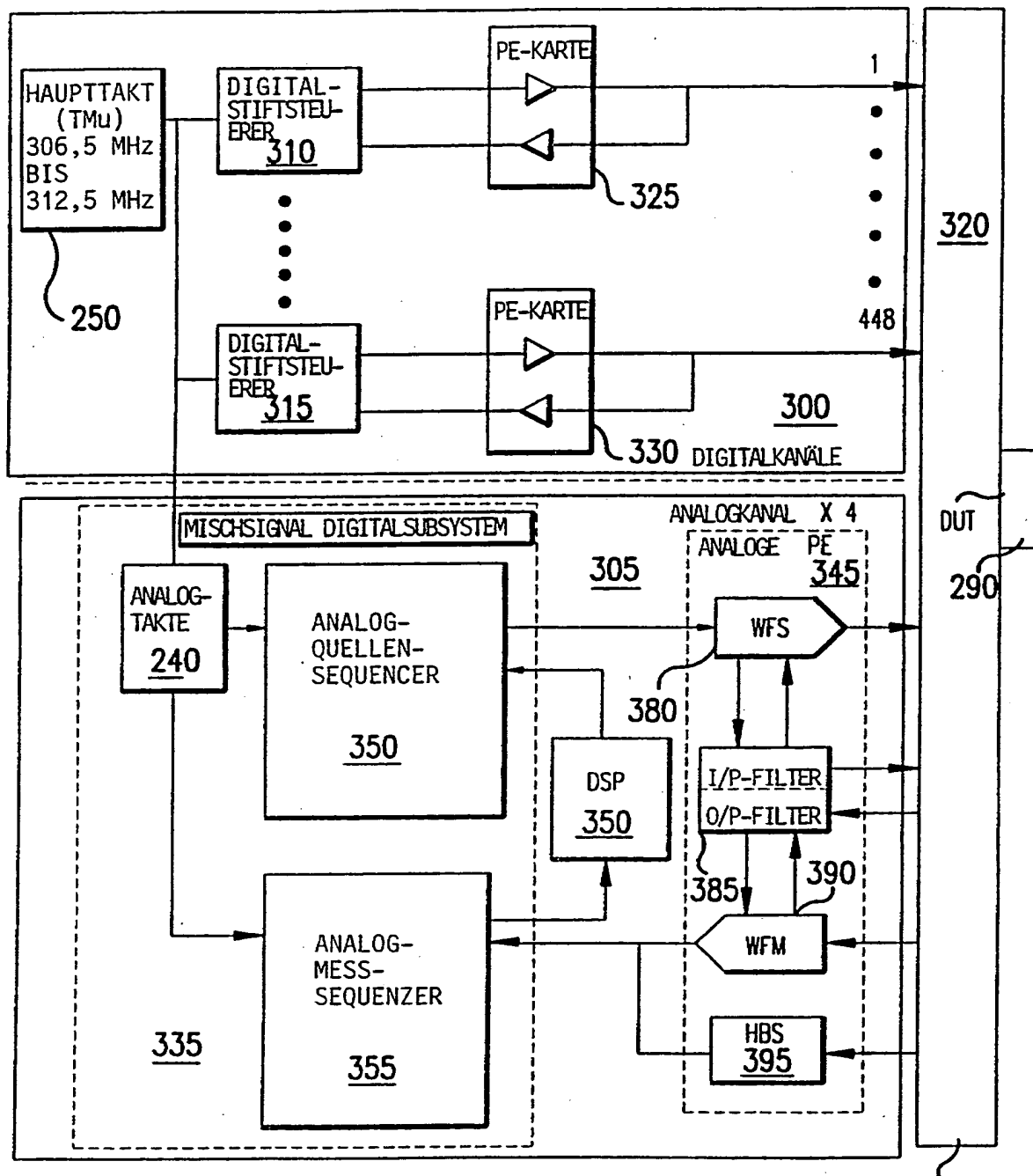


FIG.3

LASTPLATINE

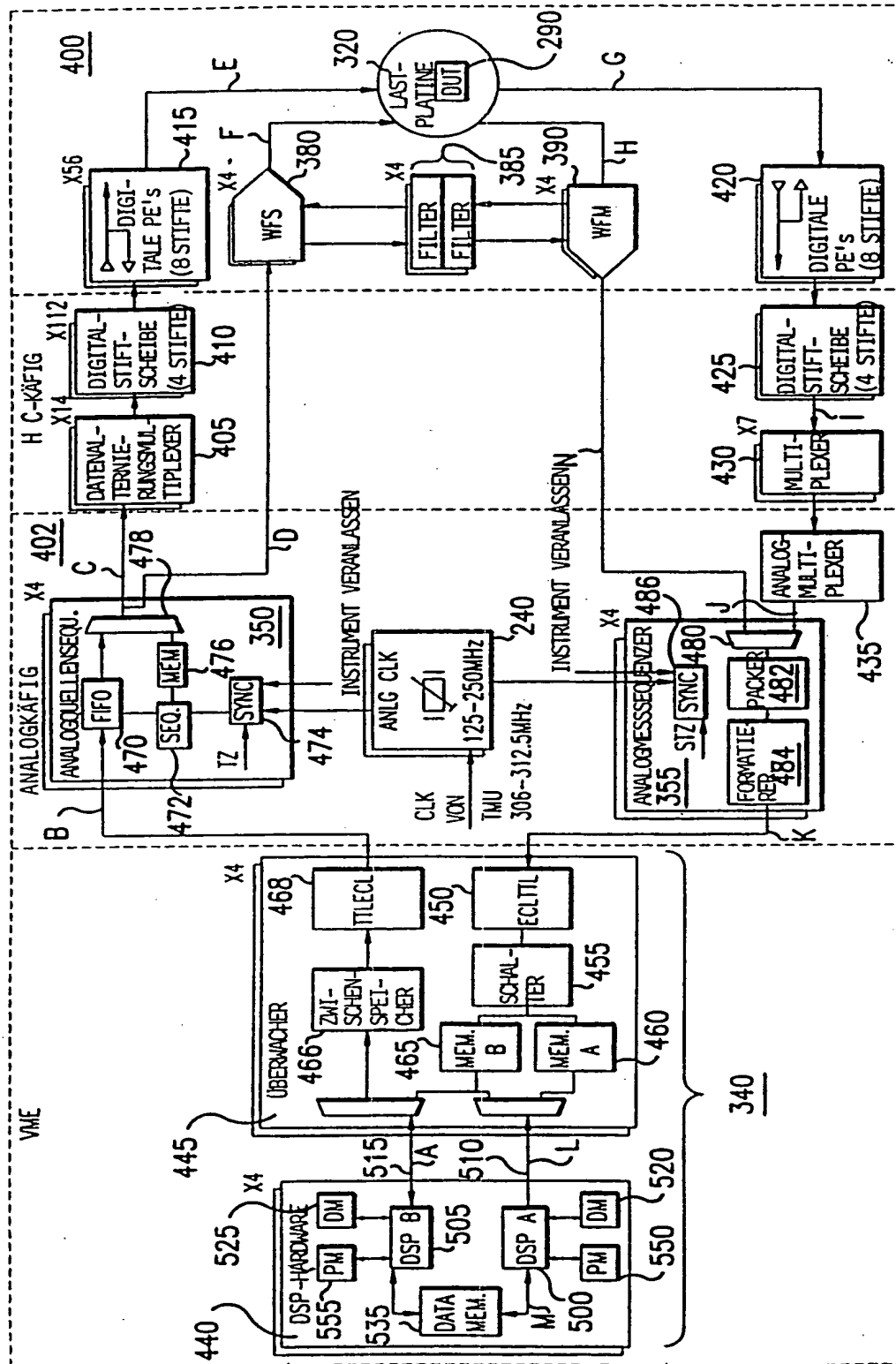


FIG. 4

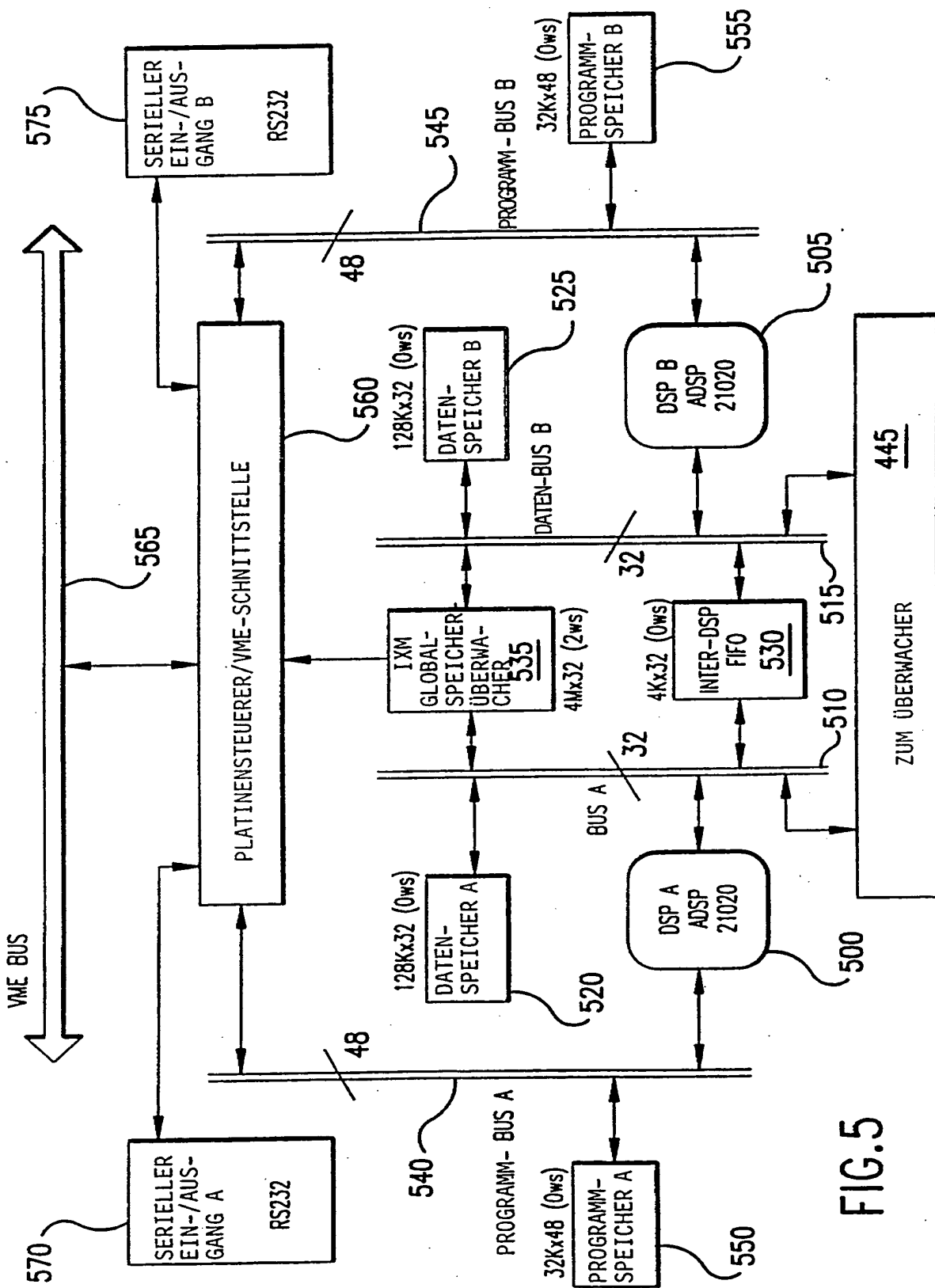


FIG. 5

FIG.6

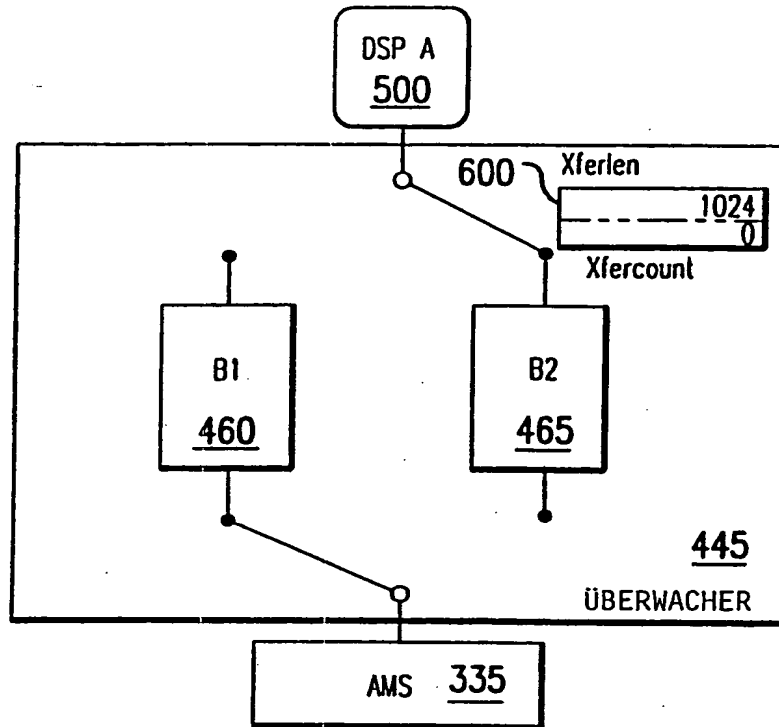


FIG.7

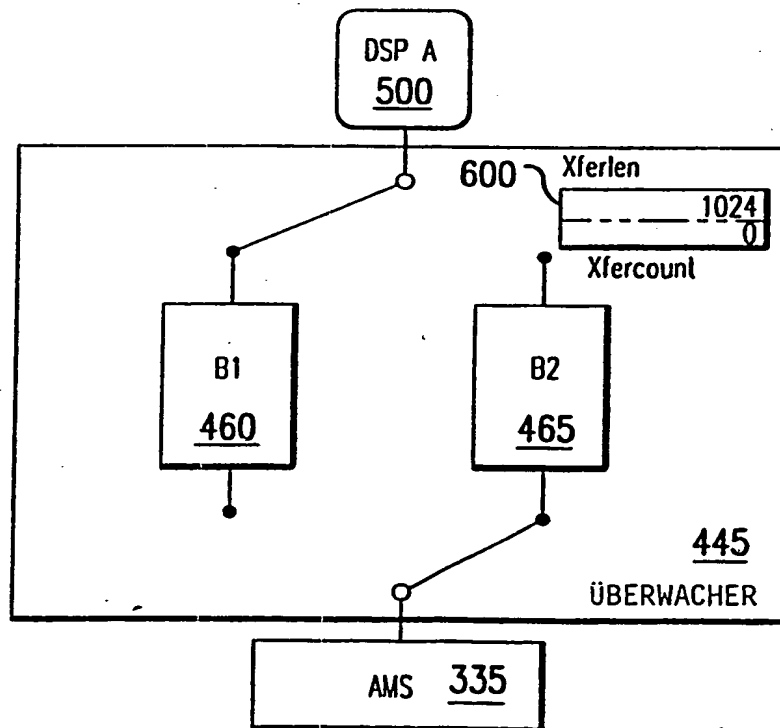
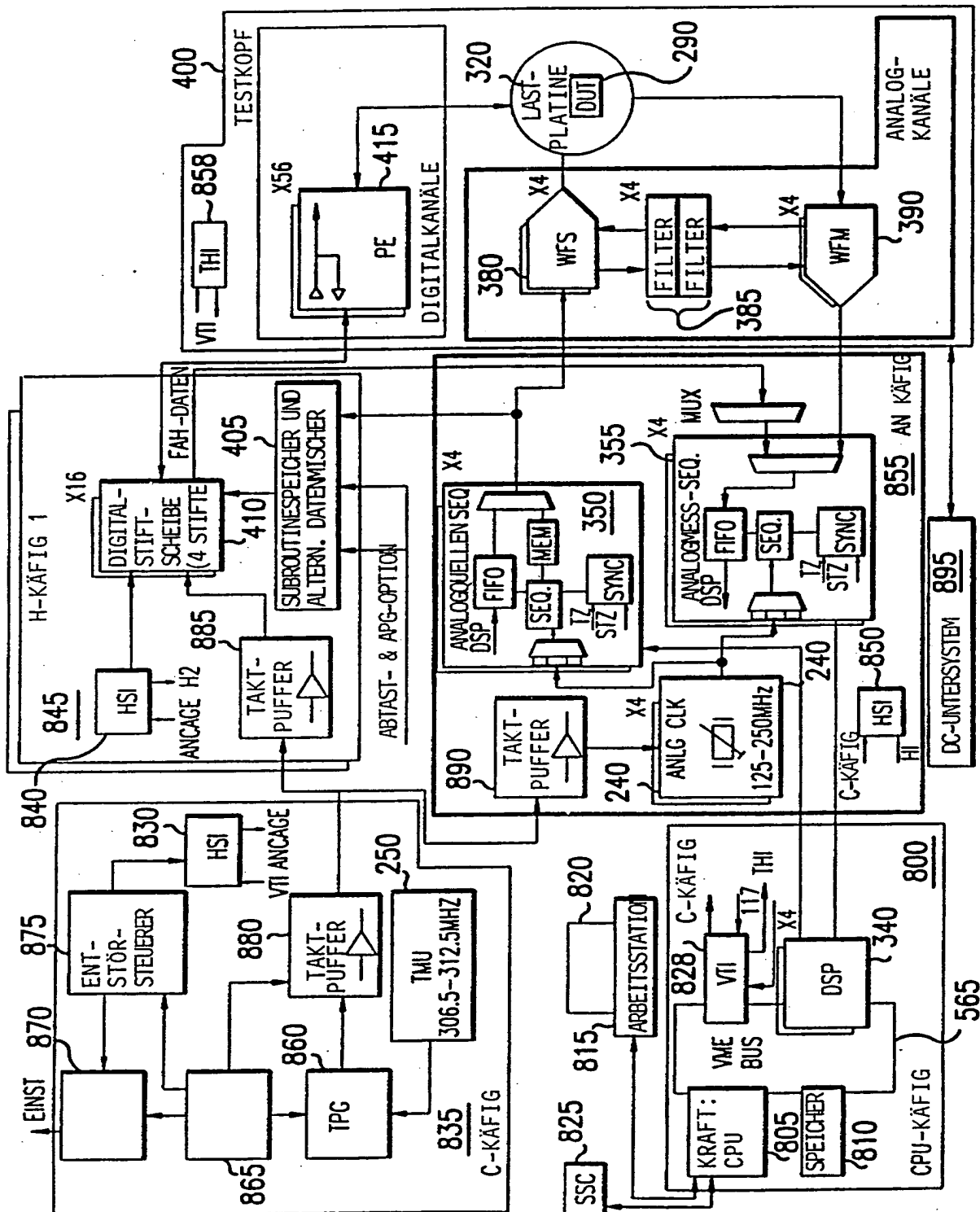


FIG.8



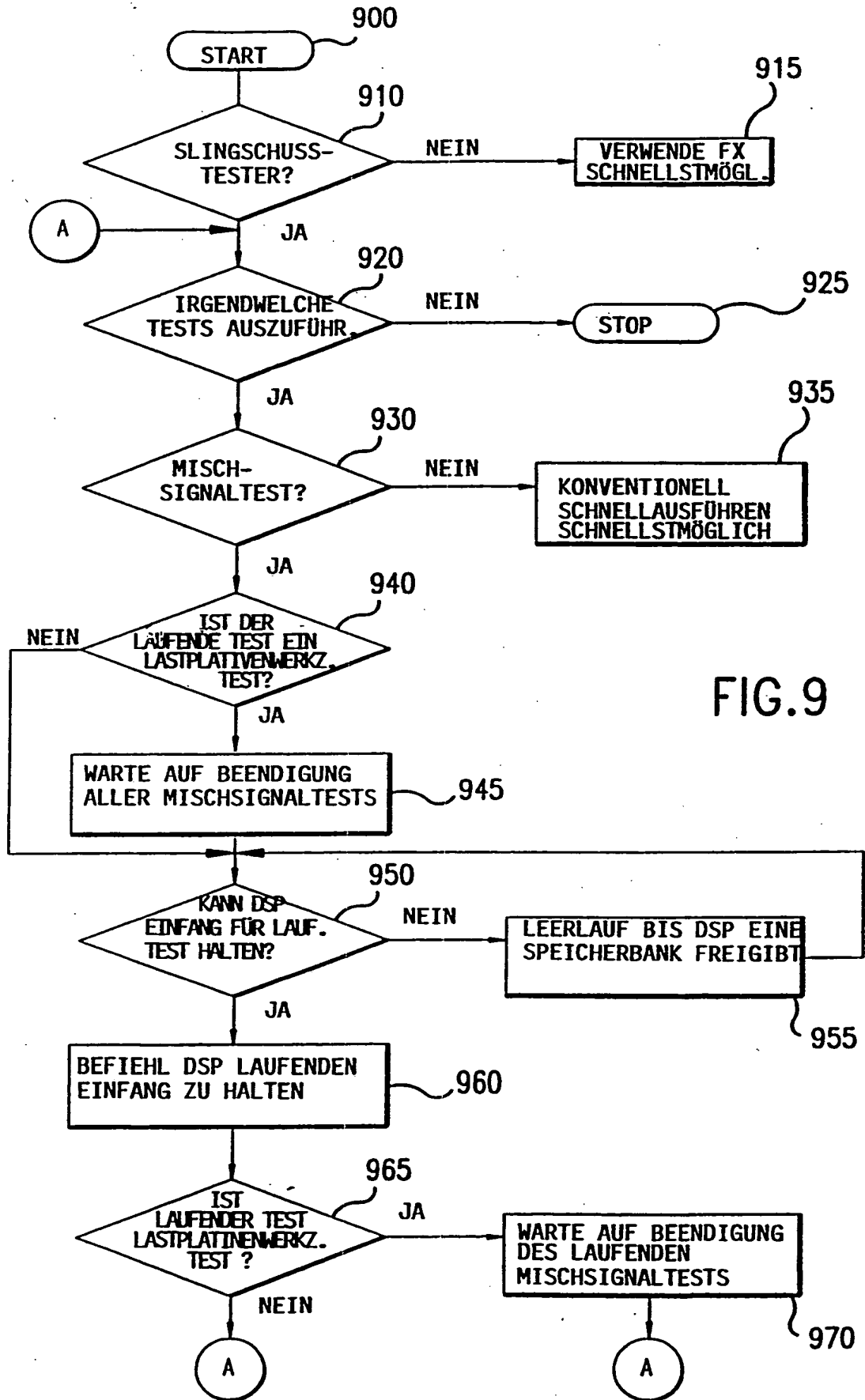
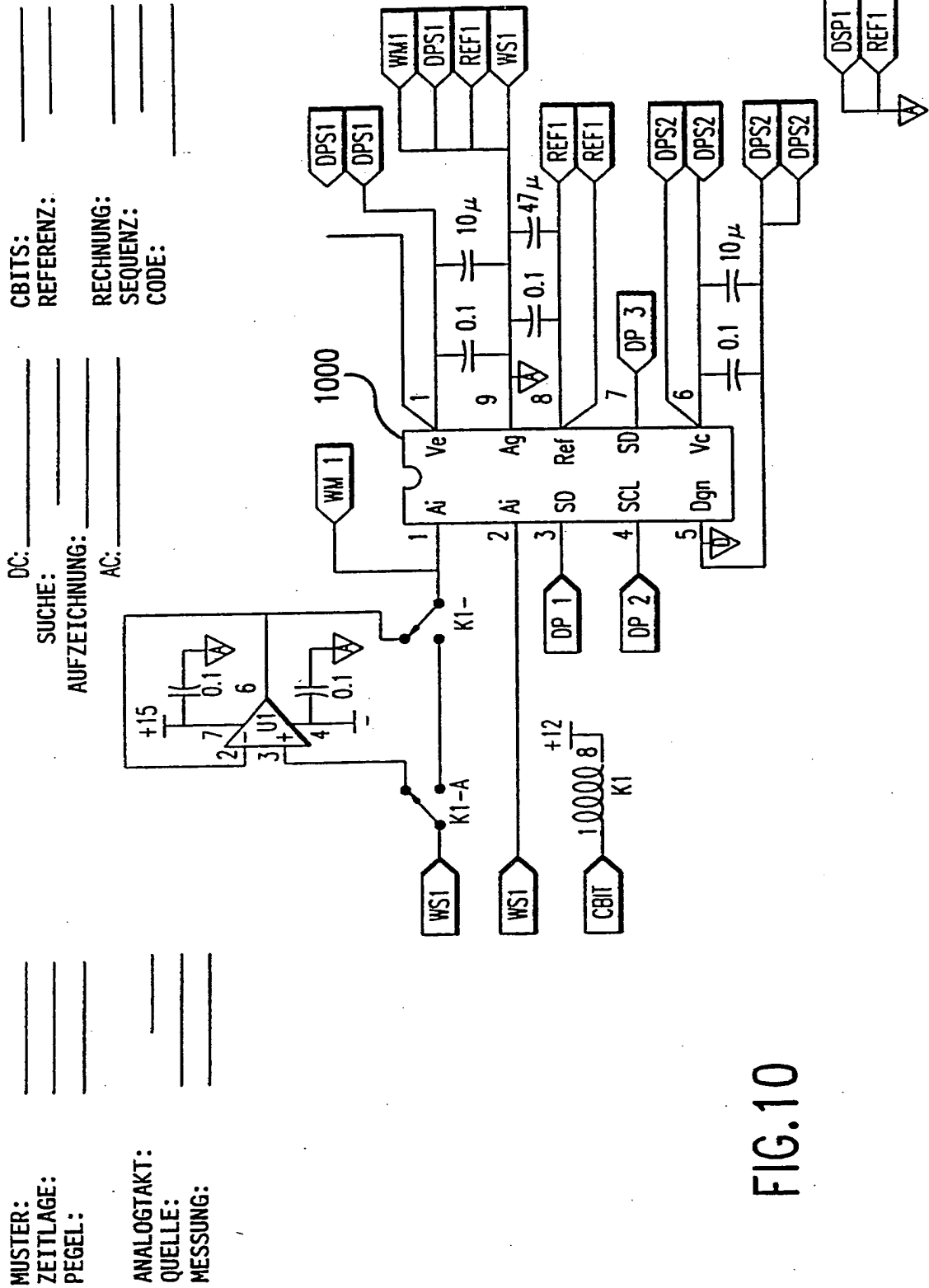


FIG.9



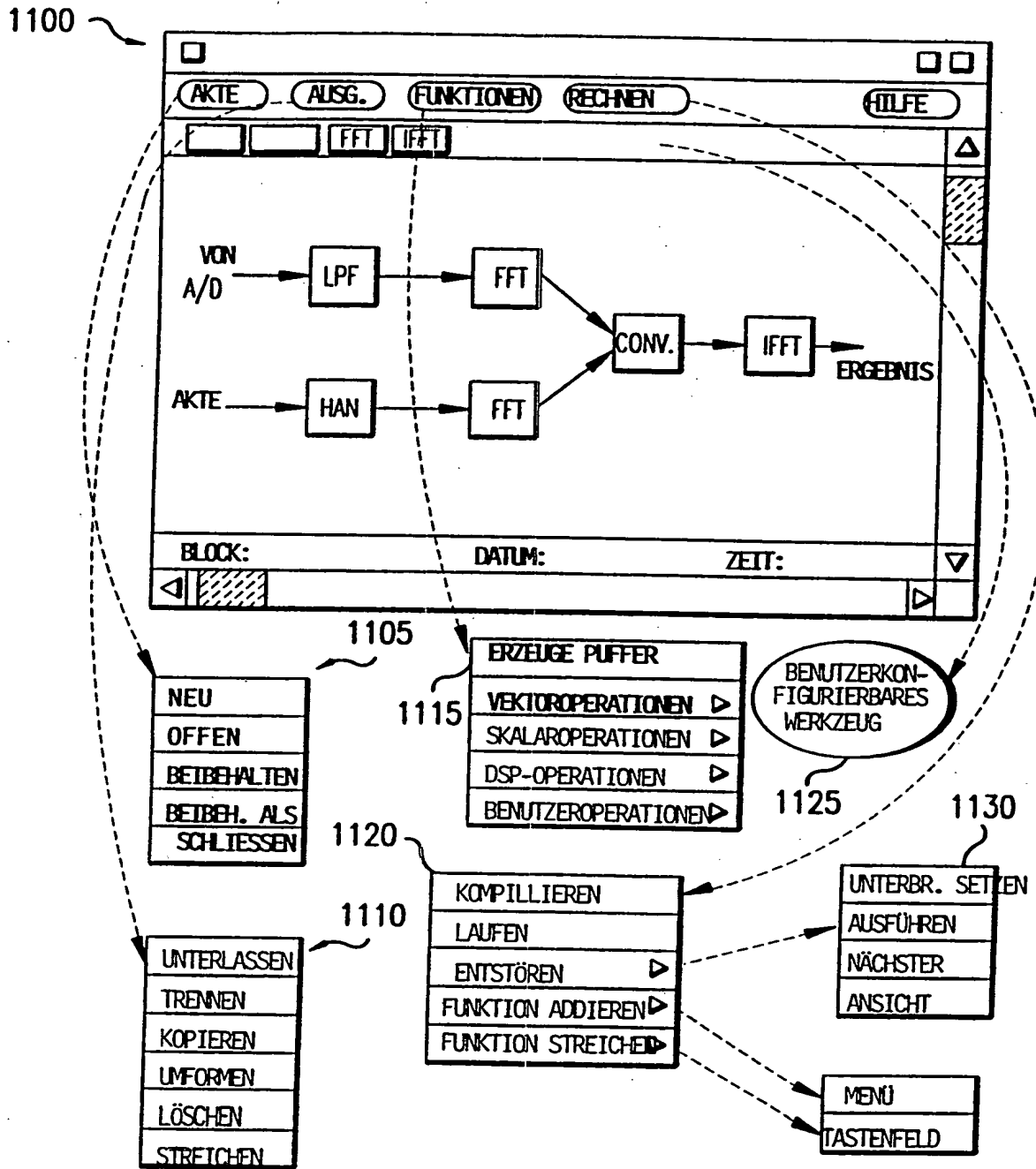


FIG.11

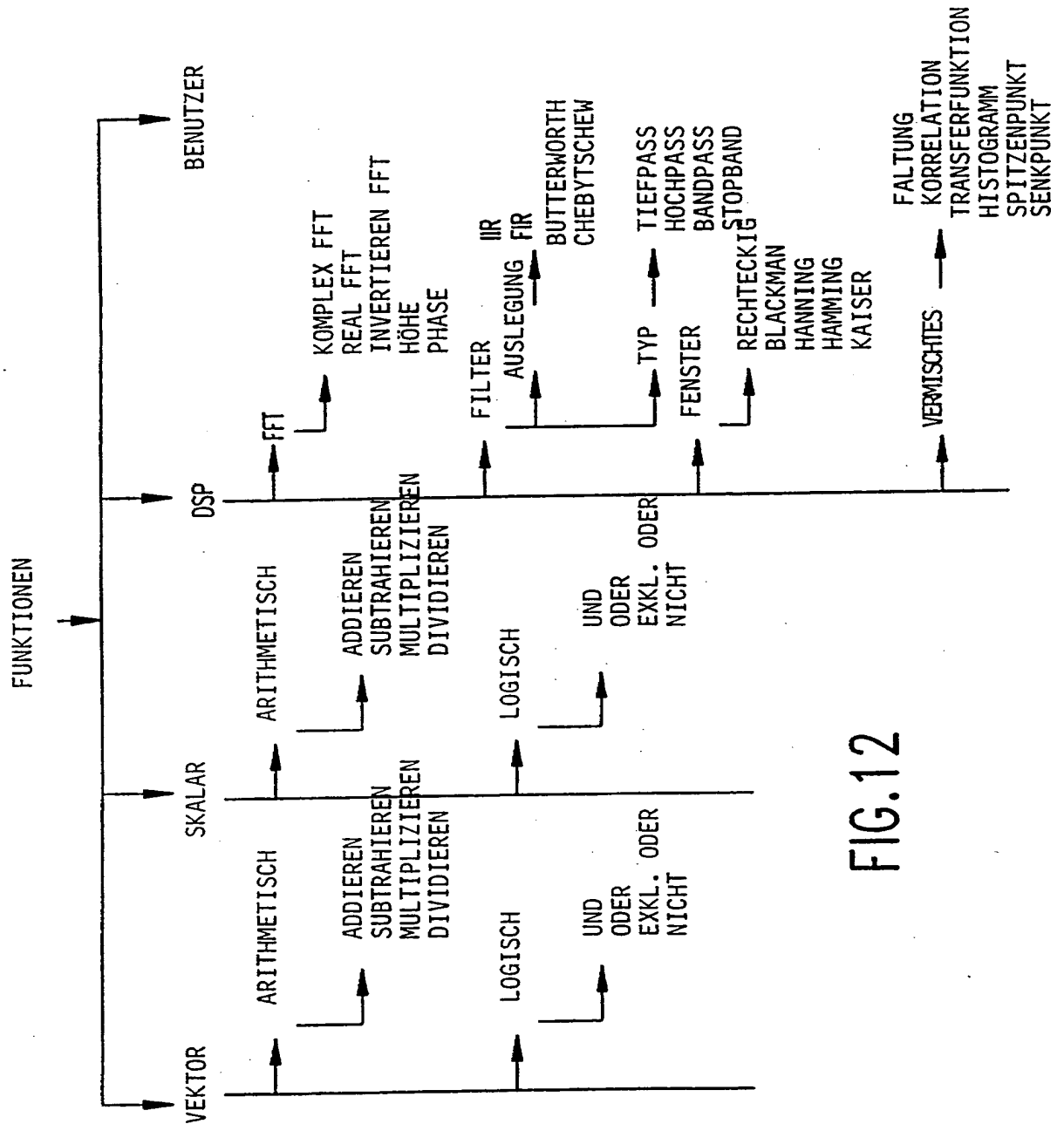


FIG.12

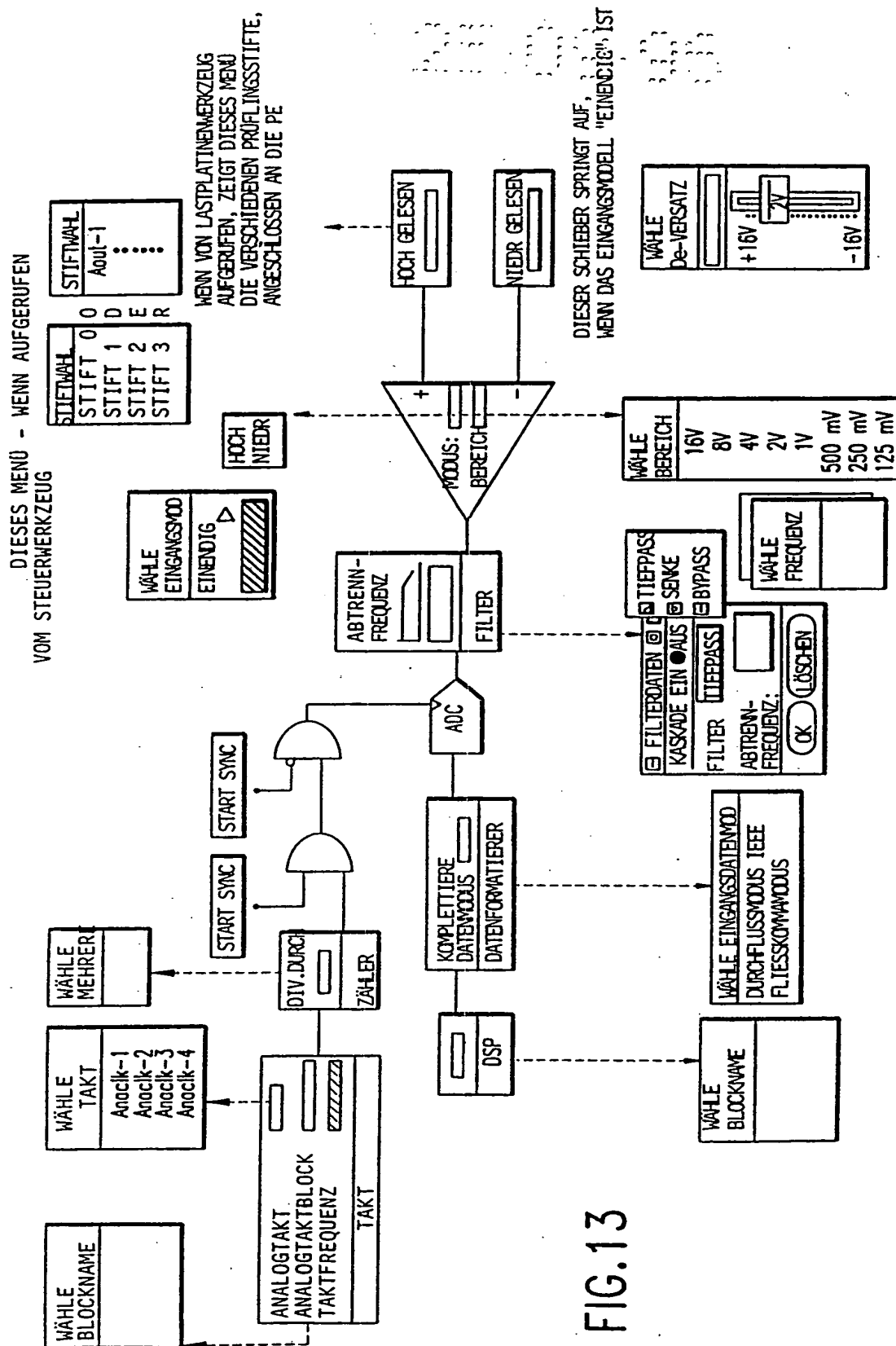


FIG. 13

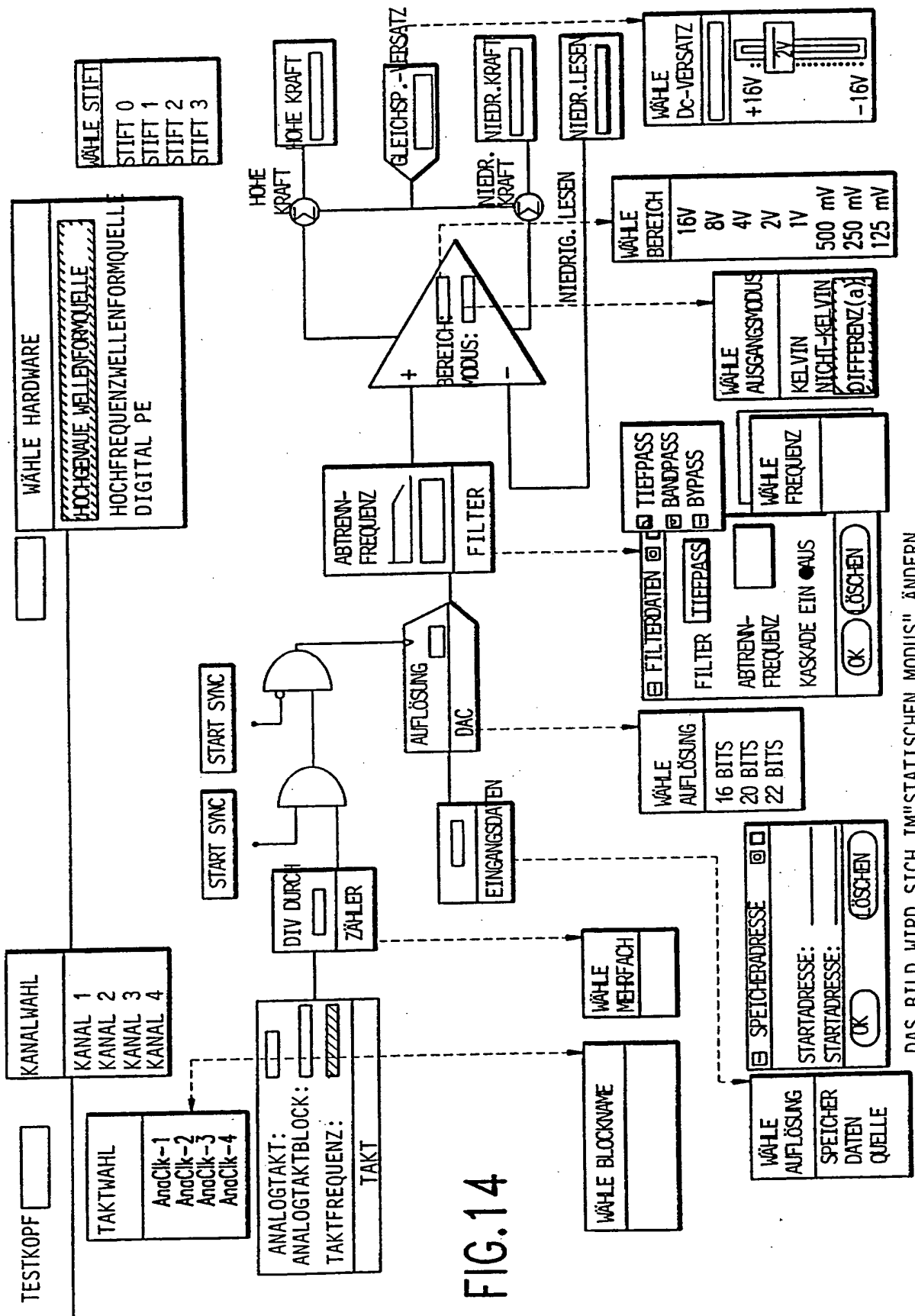


FIG. 14

**DAS BILD WIRD SICH IM"STATISCHEN MODUS" ÄNDERN
ALS NÄCHSTES GEZEIGT**

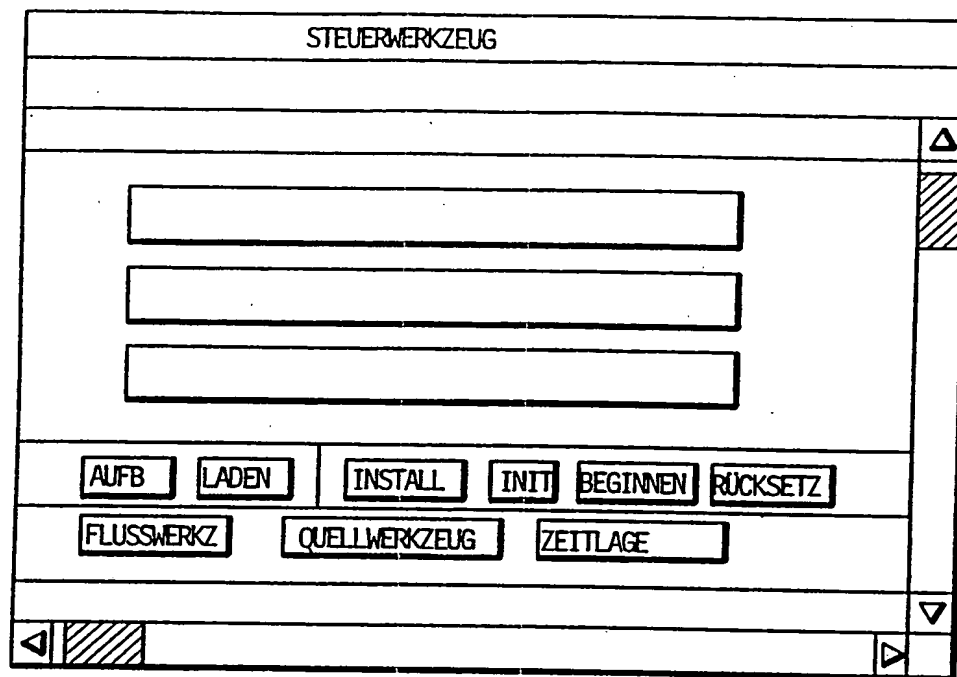


FIG. 15

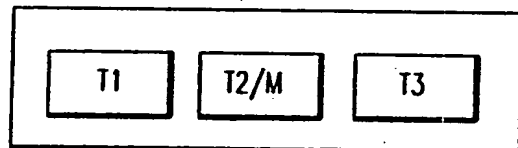


FIG. 16

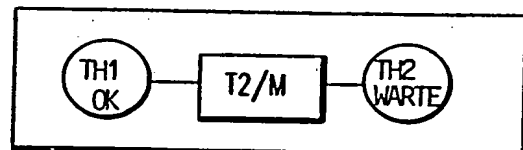


FIG. 17

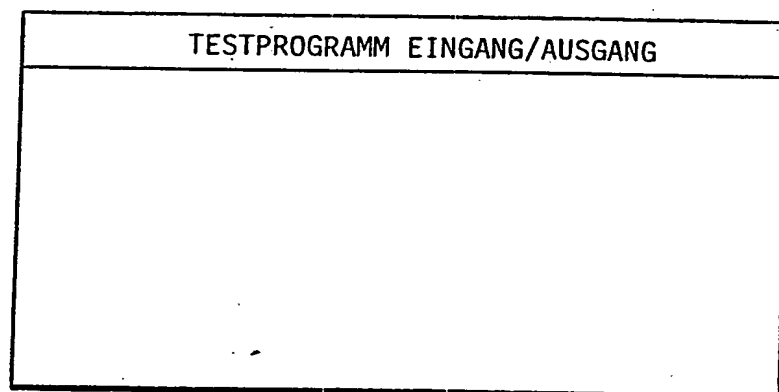


FIG. 18

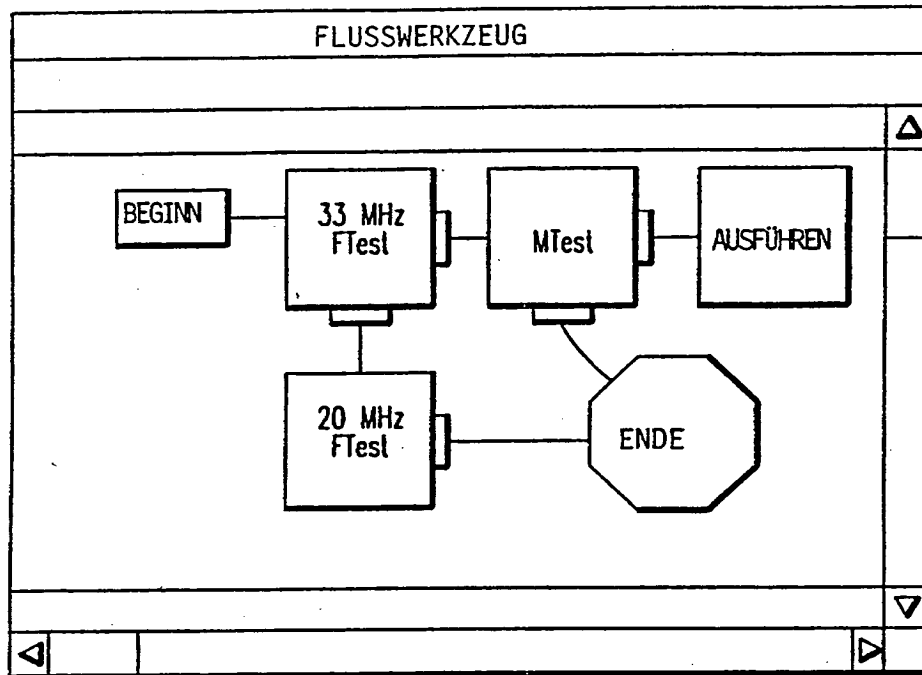


FIG.19

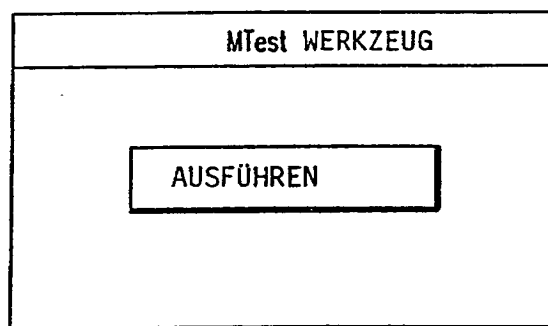
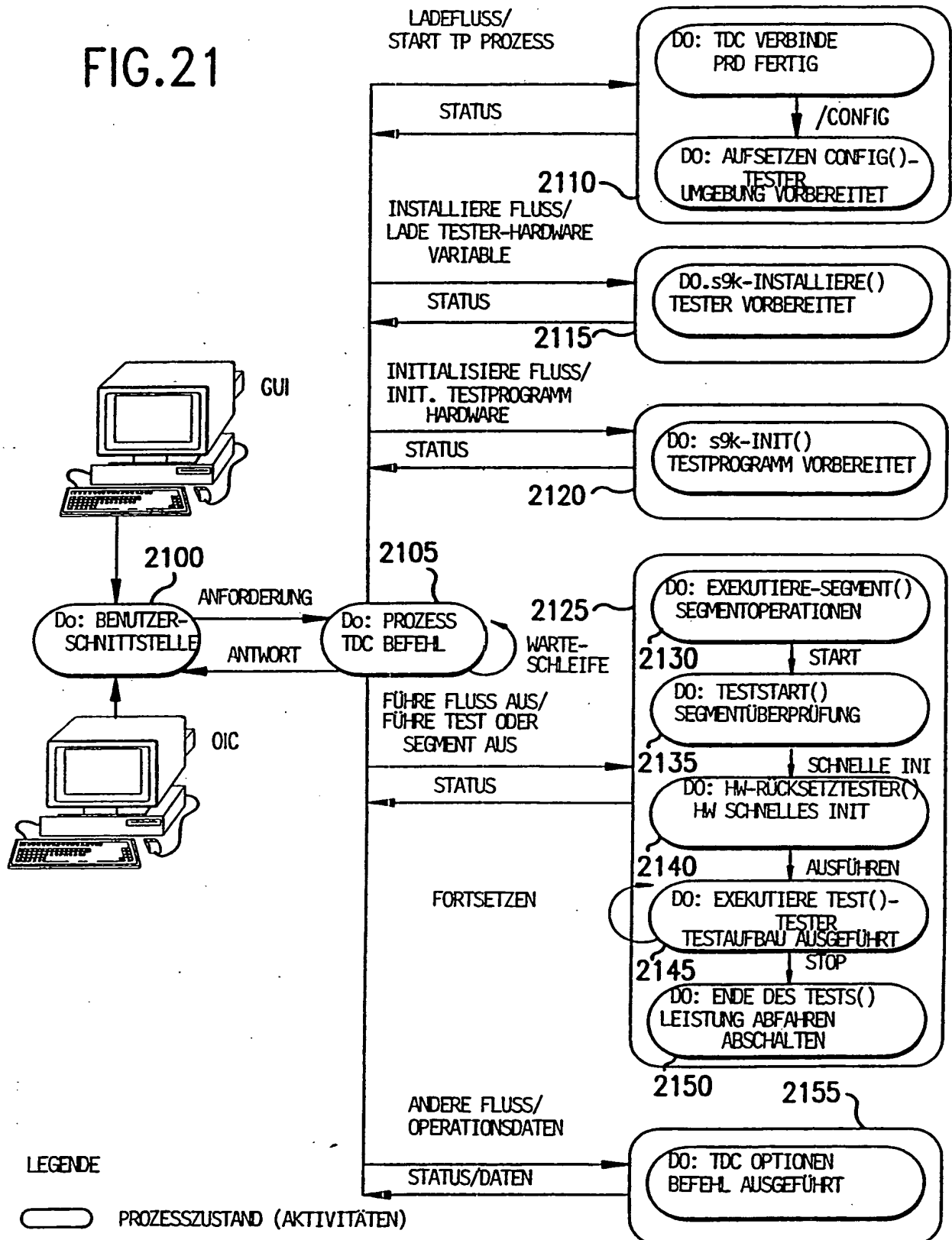


FIG.20

FIG.21



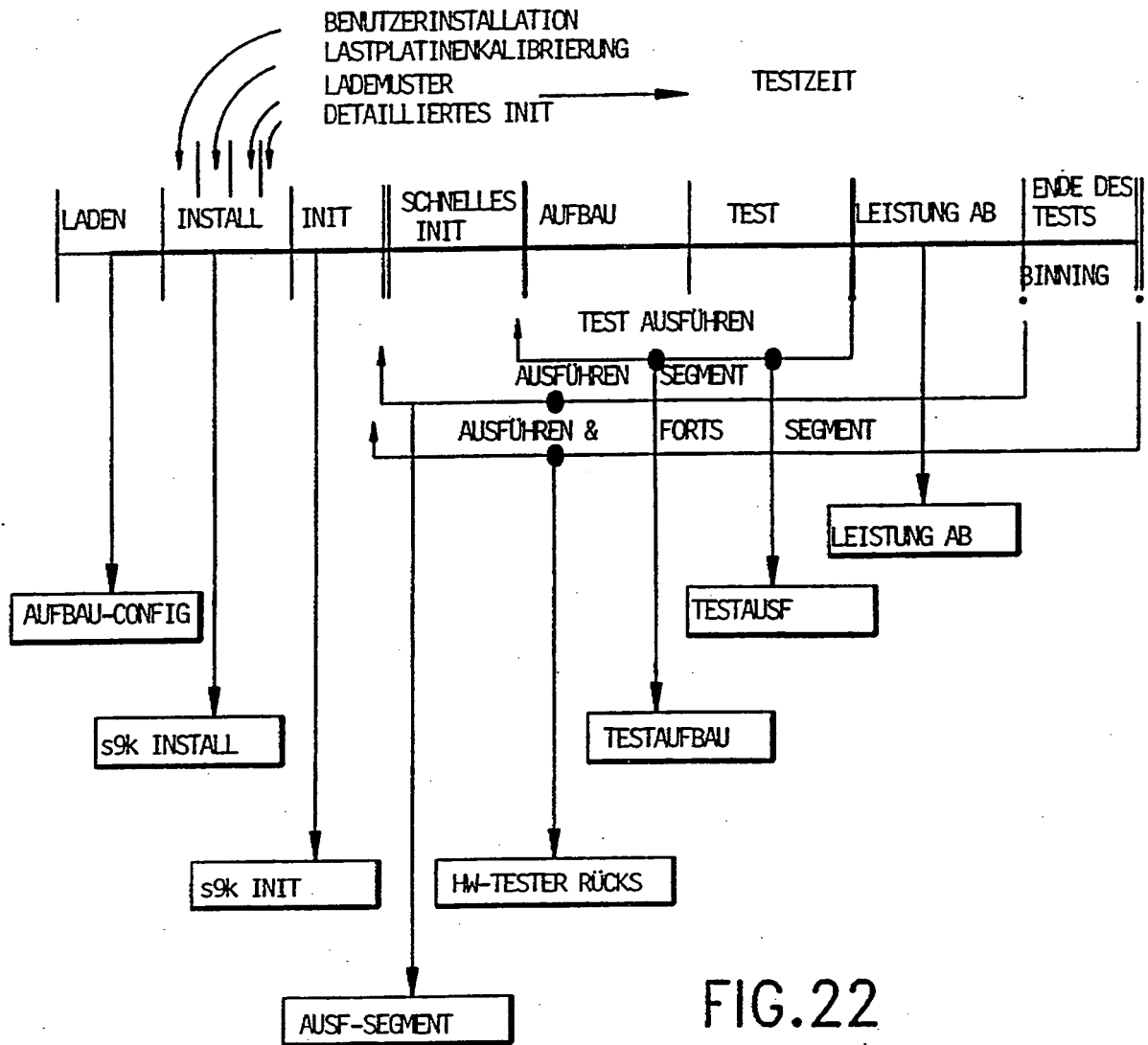


FIG.22

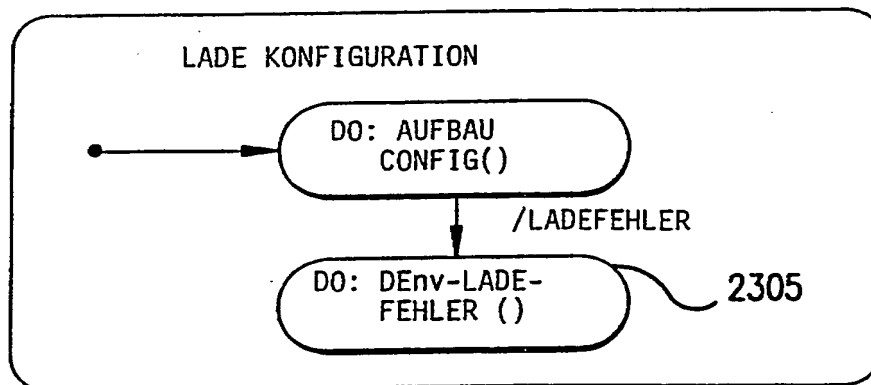


FIG.23

FIG.24

2115

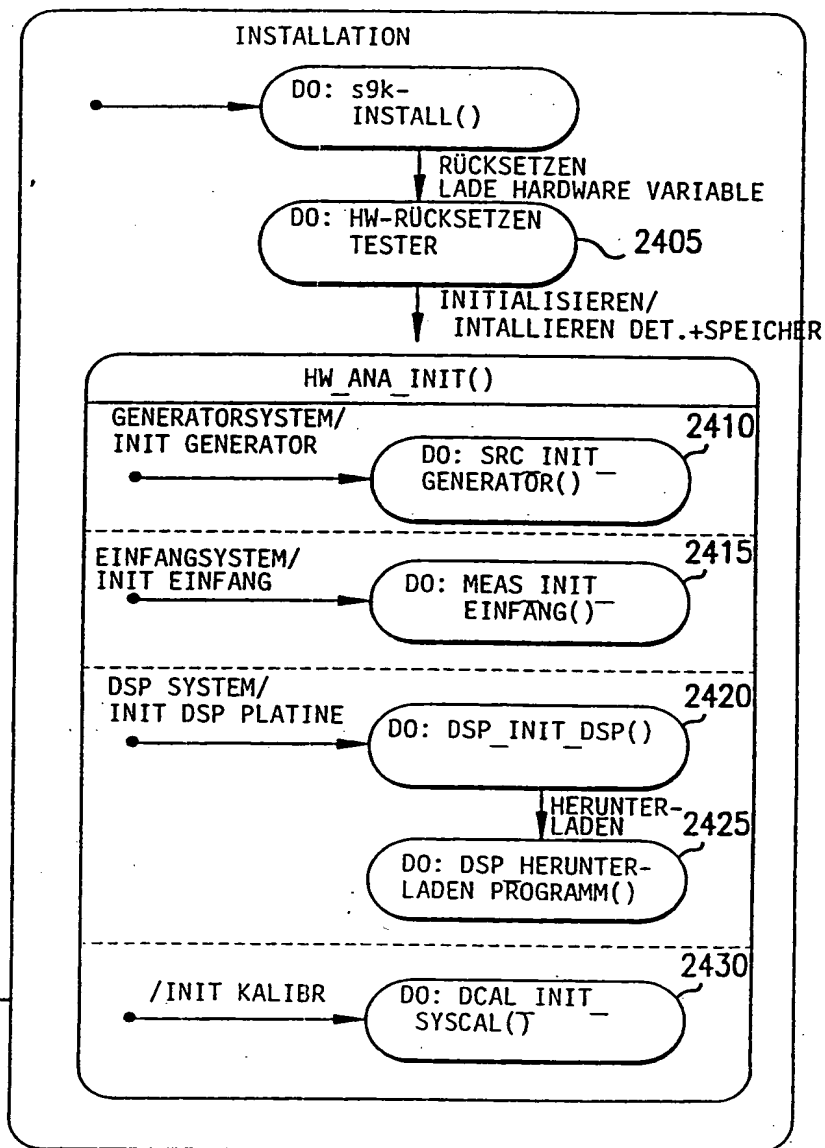


FIG.25

2125

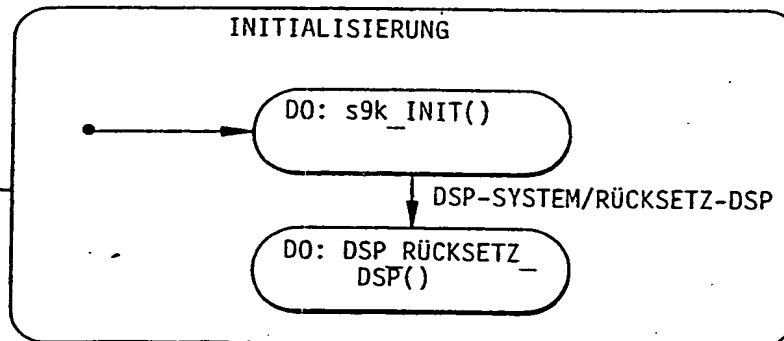
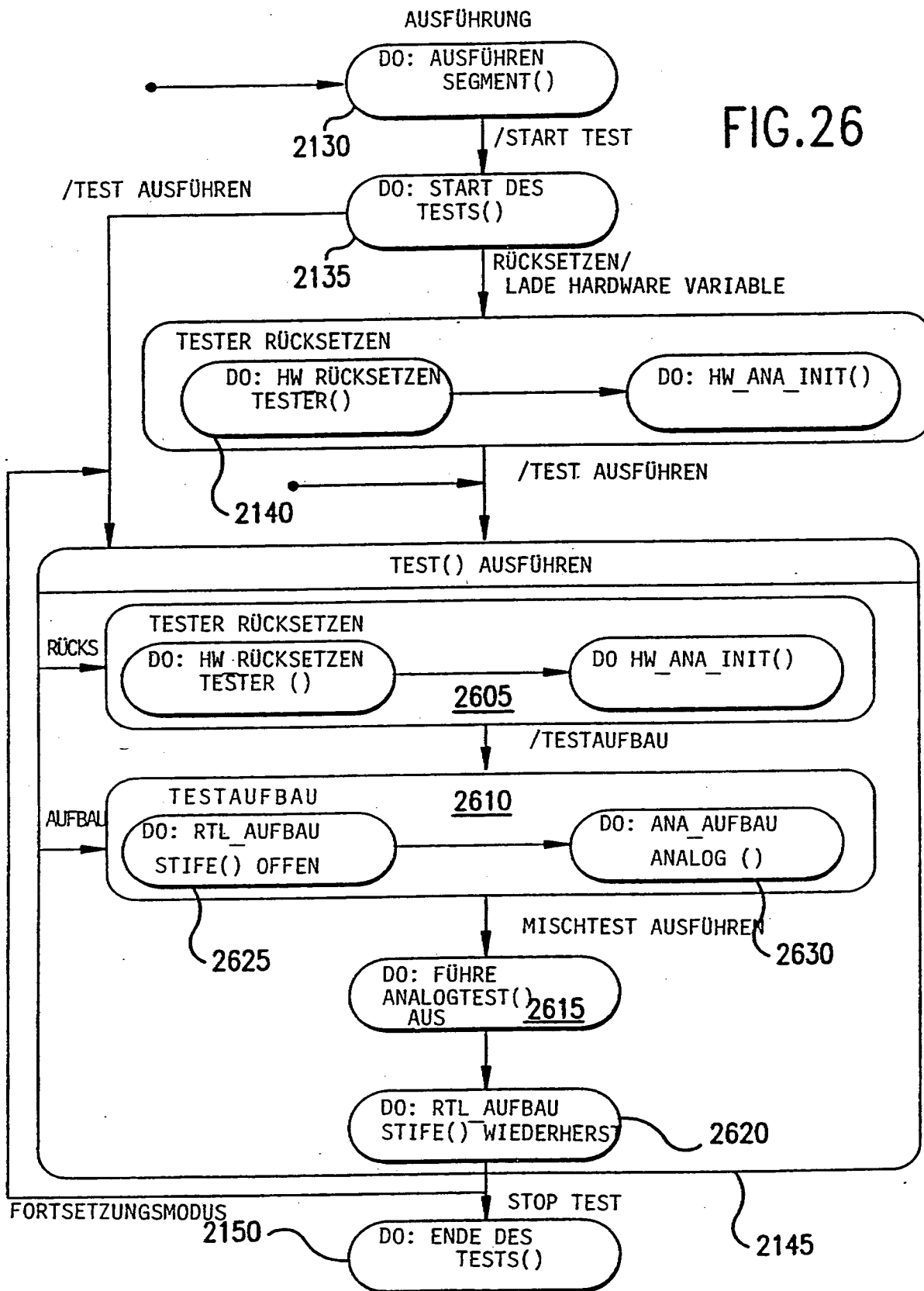


FIG.26



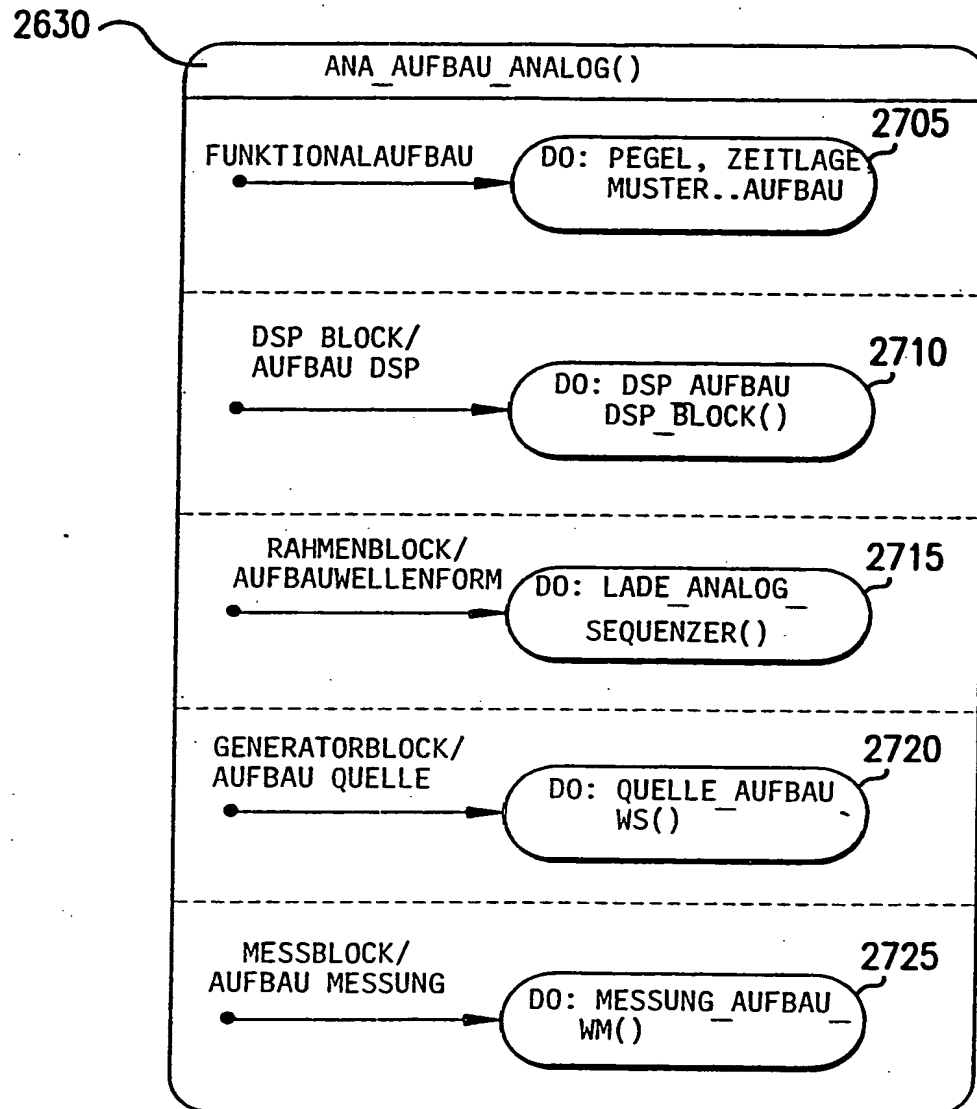
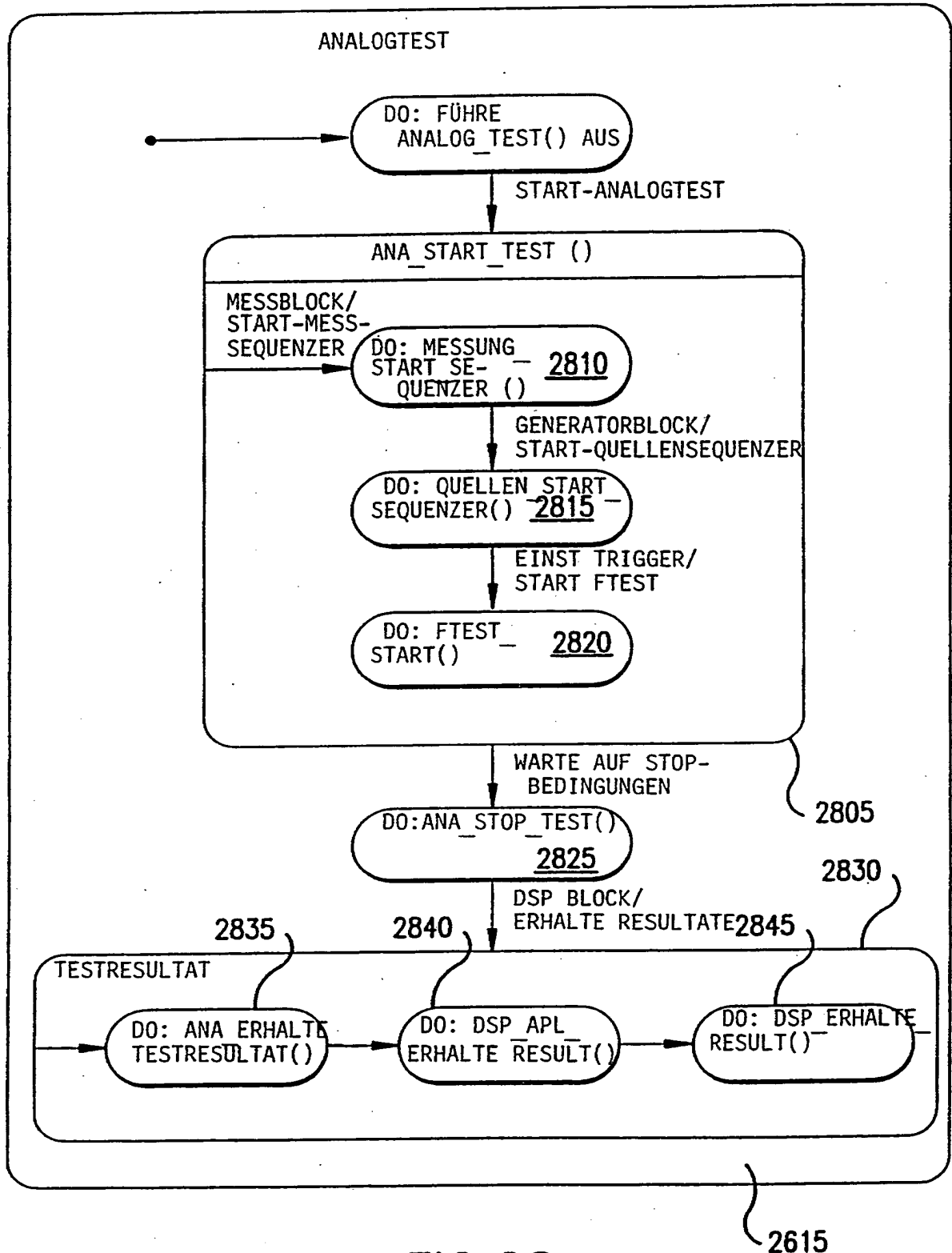


FIG.27



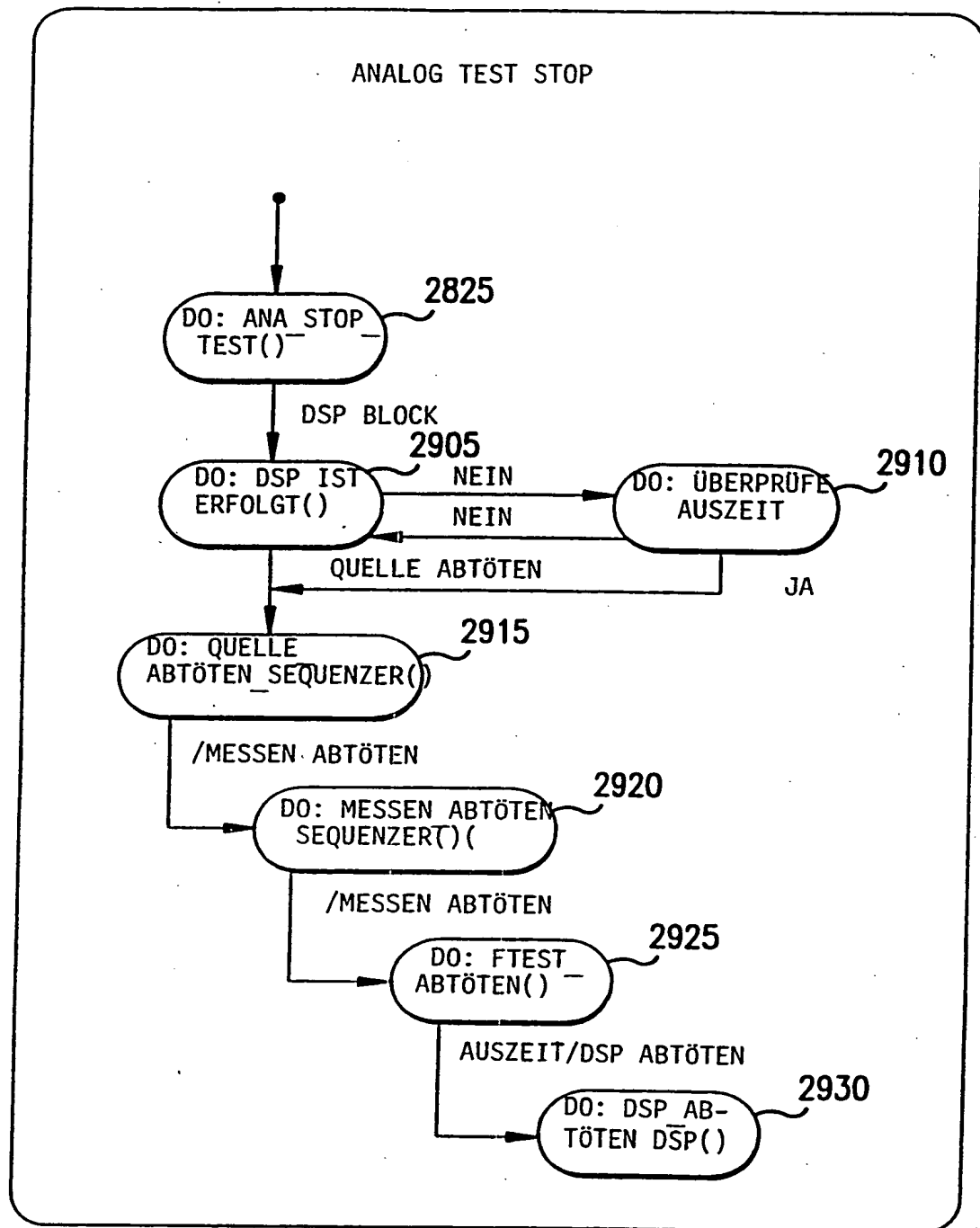


FIG.29

CODEC Tx/Rx SNR TEST

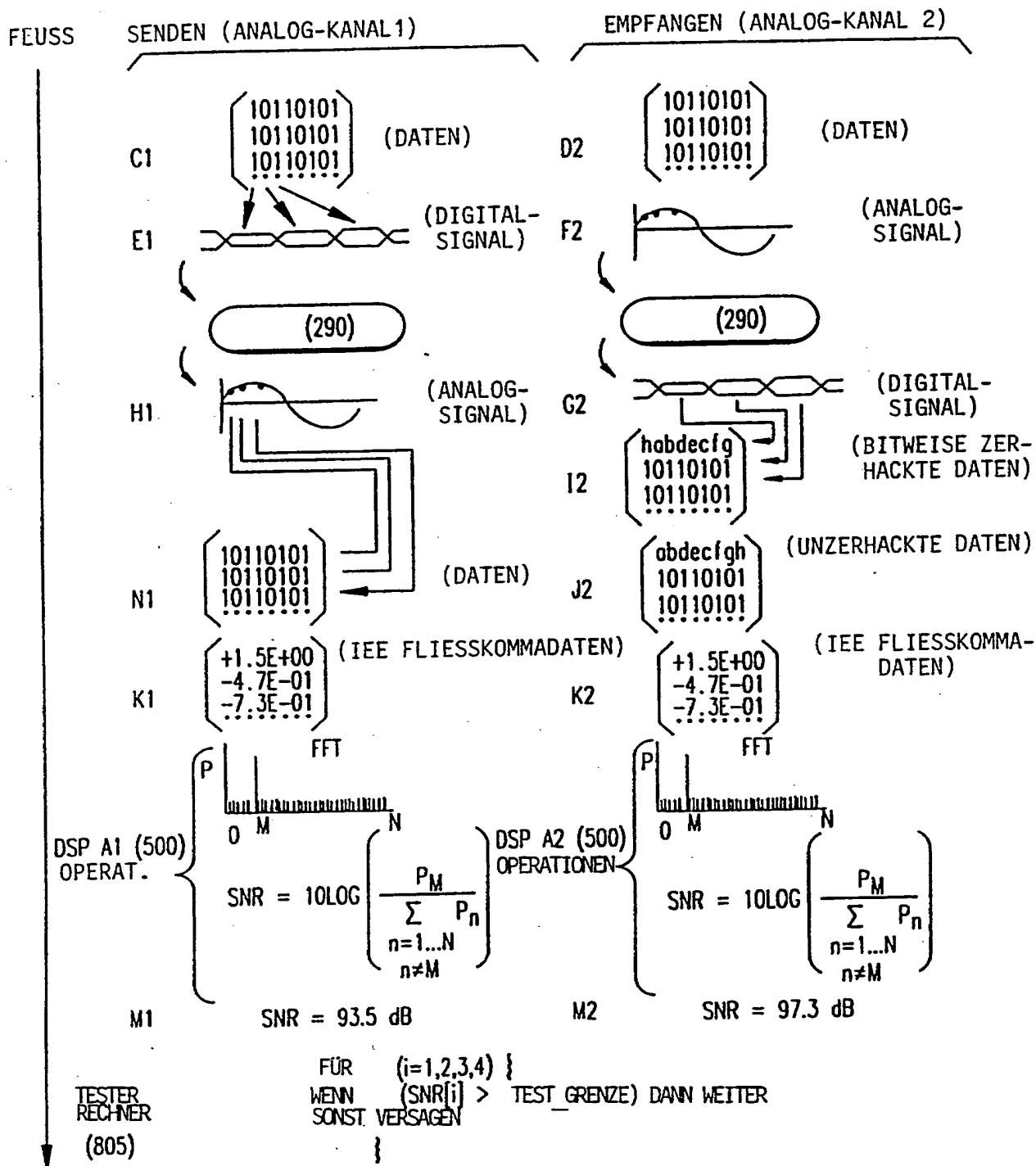
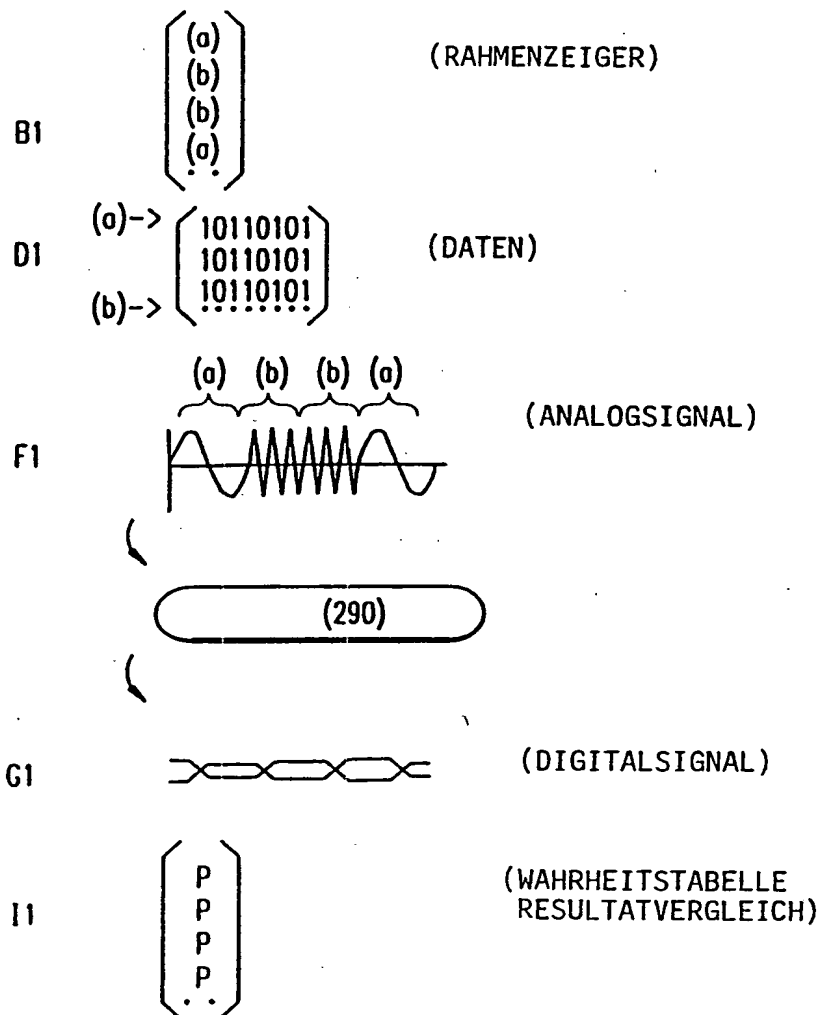


FIG.30

MODEM-BIT-FEHLERTEST

FLUSS



TESTER
COMPUTER
(805)

WENN (KEIN VEKTORVERSAGEN),
DANN WEITER
SONST FEHLER

FIG.31

ANALOG/DIGITAL-REGELSCHLEIFENCODE FÜR FLANKENÜBERGANGS-
GENAUIGKEITSTEST

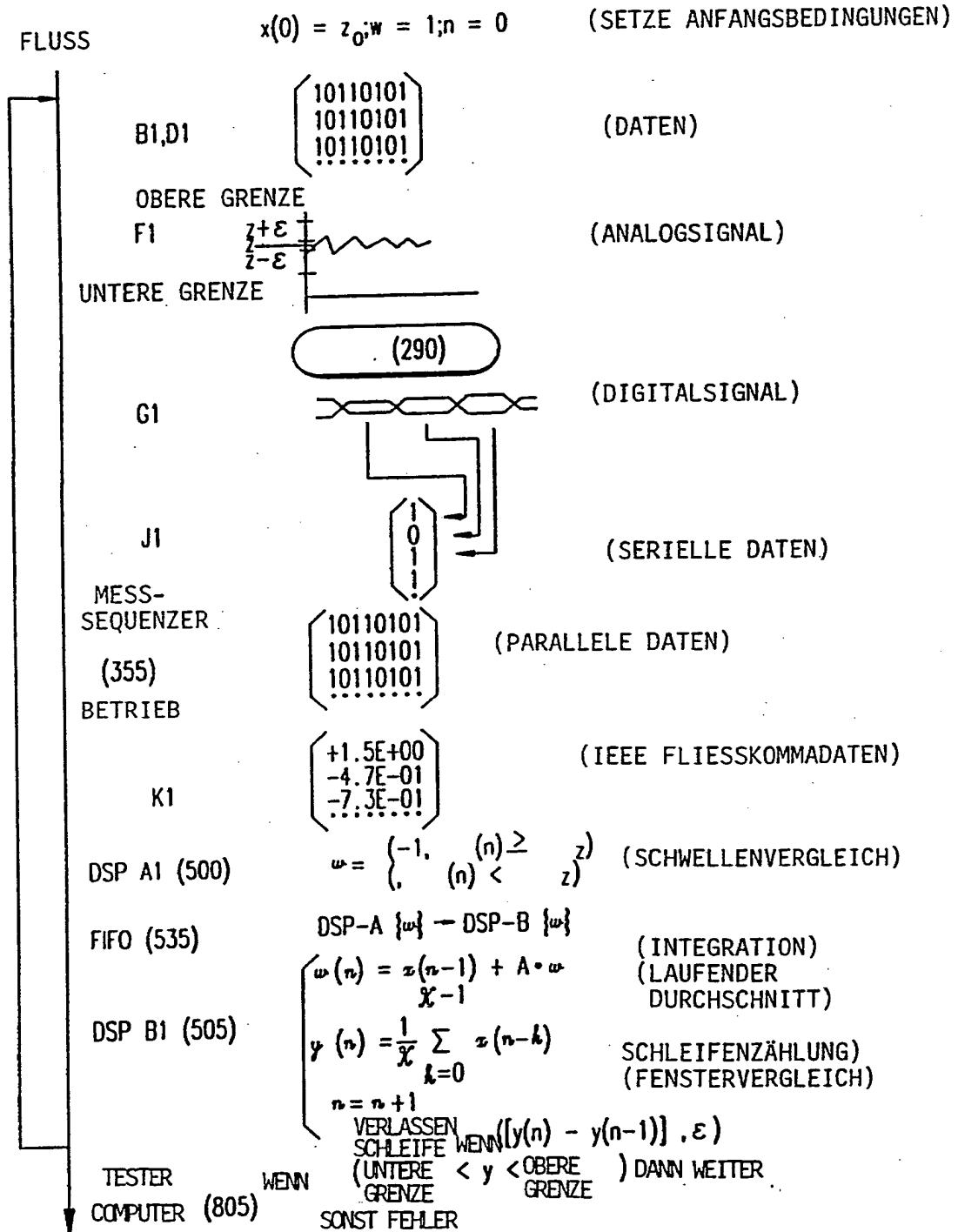


FIG.32